Computational and Structural Biotechnology Journal 23 (2024) 2779–2797

Contents lists available at ScienceDirect

Computational and Structural Biotechnology Journal

journal homepage: www.elsevier.com/locate/csbj

Review Article

Structure-based protein and small molecule generation using EGNN and diffusion models: A comprehensive review

Farzan Soleymani [a,*], Eric Paquet [b,c], Herna Lydia Viktor [c], Wojtek Michalowski [a]

[a] Telfer School of Management, University of Ottawa, ON, K1N 6N5, Canada
[b] National Research Council, 1200 Montreal Road, Ottawa, ON, K1A 0R6, Canada
[c] School of Electrical Engineering and Computer Science, University of Ottawa, ON, K1N 6N5, Canada

ARTICLE INFO

Keywords:
Diffusion models
Protein-backbone generation
Generative models
Graph representation
Equivariant graph neural network
Protein sequence

ABSTRACT

Recent breakthroughs in deep learning have revolutionized protein sequence and structure prediction. These advancements are built on decades of protein design efforts, and are overcoming traditional time and cost limitations. Diffusion models, at the forefront of these innovations, significantly enhance design efficiency by automating knowledge acquisition. In the field of *de novo* protein design, the goal is to create entirely novel proteins with predetermined structures. Given the arbitrary positions of proteins in 3-D space, graph representations and their properties are widely used in protein generation studies. A critical requirement in protein modelling is maintaining spatial relationships under transformations (rotations, translations, and reflections). This property, known as equivariance, ensures that predicted protein characteristics adapt seamlessly to changes in orientation or position. Equivariant graph neural networks offer a solution to this challenge. By incorporating equivariant graph neural networks to learn the score of the probability density function in diffusion models, one can generate proteins with robust 3-D structural representations. This review examines the latest deep learning advancements, specifically focusing on frameworks that combine diffusion models with equivariant graph neural networks for protein generation.

1. Introduction

Proteins are biological macromolecules that assume crucial roles in cellular processes. The interaction between a protein's backbone, side chains, and surrounding environment collectively shapes its 3-D structure, dictating its biological role [1]. Protein generation, the process of creating new proteins, is crucial in fields ranging from medicine to biotechnology [2,3]. The prospect of computationally generating new, physically foldable protein structures presents an exciting opportunity for exploring novel pathways within cellular mechanisms and potentially devising treatments for presently untreatable diseases. Traditional methods typically rely on heuristics to reconstruct fragments from experimentally characterized proteins [4,5]. However, these methods are constrained by cost, existing knowledge, and available data [6]. The sheer number of possible protein sequences poses a challenge to discovering novel proteins with desired properties such as high binding affinity and manufacturability [7,8]. This is further complicated by the multitude of potential conformations of the peptide backbone, along with the need to ensure accuracy in the geometry of chemical bonds and interactions.

The primary structure of a protein, characterized by its unique sequence of amino acids, dictates the identity and position of each residue side chain in the protein backbone. The backbone can be viewed as the result of intramolecular interactions occurring along the linear chain, directing the folding process towards a distinct 3-D conformation [1,9]. Accurately describing a protein's structure relies on precisely determining the 3-D coordinates for both the alpha carbon (C_α), the first carbon in the residue as depicted in Fig. 1, and the atoms in the side chain. These coordinates are influenced by various factors related to the structural characteristics and interactions within the protein. These factors limit how atoms can be spatially arranged, and include steric hindrance, chemical properties, secondary structure propensities, hydrogen bonding patterns, hydrophobic interactions, and evolutionary conservation [10]. By incorporating these factors into computational models, we can improve the accuracy and reliability of protein structure prediction, ensuring that the predicted structures closely follow the fundamental

* Corresponding author.
E-mail addresses: fsoleyma@uottawa.ca (F. Soleymani), Eric.Paquet@nrc-cnrc.gc.ca (E. Paquet), hviktor@uottawa.ca (H.L. Viktor), wojtek@telfer.uottawa.ca (W. Michalowski).

https://doi.org/10.1016/j.csbj.2024.06.021
Received 19 April 2024; Received in revised form 13 June 2024; Accepted 18 June 2024
Available online 26 June 2024

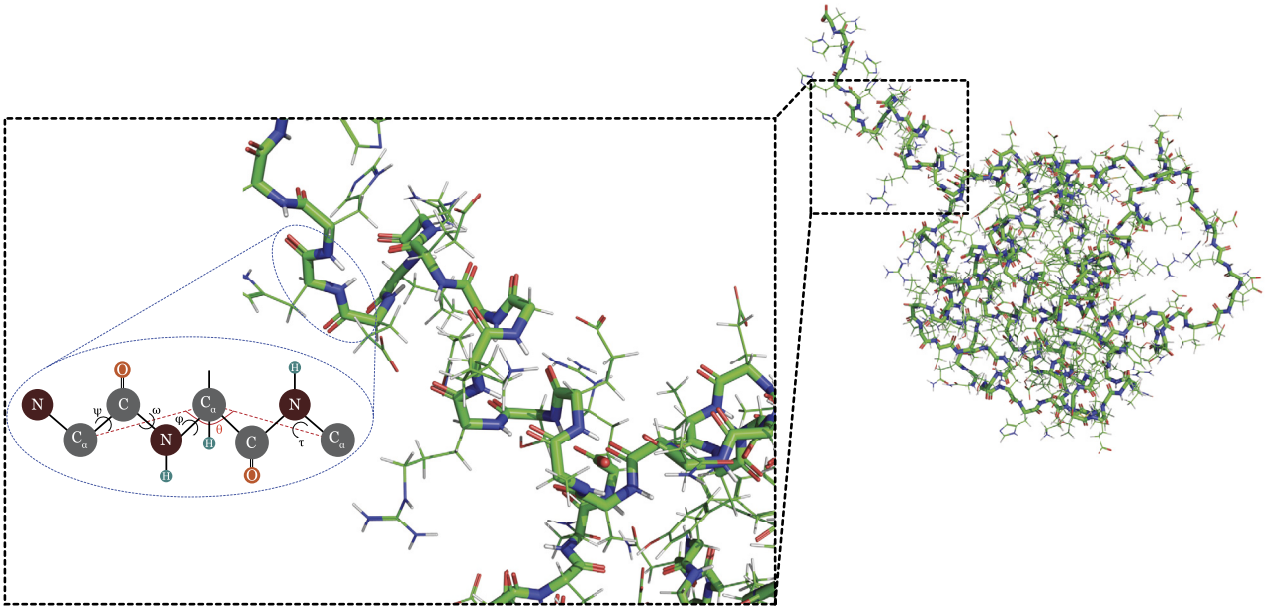

Fig. 1. The protein backbone of chain A in PDB: 1bxl. This image highlights three residues, each consisting of an alpha carbon (C_α) linked to an amino group (NH_2) and a carboxyl group (COOH), with the backbone and its angles magnified for clarity [13].

principles guiding protein folding and stability [11]. Given that the overall geometric attributes of proteins are predominantly determined by their backbones, the process of generating protein structures can be simplified to the generation of backbones, often sourced from either a natural or artificially created protein (i.e. *de novo* design). Subsequently, one must find a sequence of amino acids that will fold into the established backbone structure [12].

Recent advancements in deep learning [14] methods have found a wide range of applications in computational biology and bioinformatics, such as protein structure prediction [11,15–21], protein design [22,23], unsupervised protein design [24,25], binding site prediction [26–28], and predicting protein–protein and protein–ligand interaction [9,29–36]. Despite the significant progress made in sequence design [37,38], generating protein backbones remains a challenge. Owing to the success of deep generative models in addressing similar high-dimensional modelling and inference challenges in various other domains, a notable example being the generation of photorealistic images from text [39–41], substantial efforts have been devoted to developing generative models of protein space [12,24,38,42–50], such as unsupervised sequence generation and representation learning using language models [24,47].

Diffusion models, a class of generative models based on deep learning [51–54], have exhibited superior performance, generating highly realistic data across various domains. Notable applications are found in image generation [2,41,55–60], image inpainting [61,62], speech synthesis [63], natural language processing [64–68], temporal data modelling [69–73], and multimodal modelling [39,41,55,74]. Diffusion-based generative models offer distinct advantages over other generative approaches, such as autoregressive models [75], normalizing flows [76], energy-based models [77], variational auto-encoders (VAEs) [78], and generative adversarial networks (GANs) [79]. Diffusion models are capable of learning complex data distributions, handling high-dimensional data, and generating diverse outputs [64,80–84].

Score-based generative modelling with stochastic differential equations (SDEs) combines the strengths of score-based and diffusion probabilistic models. It views the traditional stepwise addition of noise in score-based models as a continuous-time process described by an SDE [54,85,86]. This allows for efficient reversal, enabling the model to start with noise and progressively remove it to generate high-quality data, while also offering functionalities like exact likelihood calculation and controllable generation through manipulation of the starting noise. This SDE framework provides a more powerful and flexible approach to data generation [54,85,86].

Normalizing flows [76], for instance, construct flexible probability distributions by applying a series of invertible and differentiable transformations to a simple base distribution (e.g. a multivariate Gaussian). These transformations, often parameterized by neural networks, enable tractable density estimation and efficient sampling, enabling normalizing flows to model intricate, multimodal distributions and capture complex dependencies within real-world data. VAEs [78], on the other hand, learn to represent complex data distributions using probabilistic latent spaces. They employ an encoder–decoder architecture: the encoder compresses input data into a continuous latent space, learning to model the data as probability distributions. This probabilistic representation captures uncertainty and variability within the data. The decoder then samples from these learned distributions, aiming to reconstruct the original input. GANs [79], meanwhile, are unsupervised deep learning frameworks composed of two competing neural networks: a generator and a discriminator. The generator learns to produce synthetic data that mimic the distribution of a real dataset, while the discriminator attempts to distinguish between real samples and the generator's artificial outputs. This adversarial game drives both networks to improve iteratively.

In recent years, diffusion models have found application in computational biology, including protein design and generation [6,38,42,44,86–90], drug and small molecule design [84,91–96], and protein–ligand interaction modelling [97–100]. Notably, diffusion models can effectively handle high-dimensional data with diverse and scalable properties [2].

Generative adversarial networks (GANs), variational autoencoders (VAEs), and diffusion models each offer distinct advantages and face unique challenges in generative modelling. GANs are renowned for their ability to produce high-quality, realistic samples quickly after training [101,102]. This characteristic makes them particularly suitable for applications in which speed and visual fidelity are crucial. However, GANs often encounter a significant problem known as mode collapse, in which the model generates a limited variety of outputs. Moreover, their training process requires careful balancing to maintain stability [103].

In contrast, VAEs provide a more stable training framework and offer a structured latent space, enabling users to manipulate the generated data in meaningful ways [104,105]. This makes VAEs an excellent

Table 1
Comparative Trade-Offs Among Generative Model Schemes: GANs, VAEs, and Diffusion Models.

Feature	GANs	VAEs	Diffusion Models
Sample Quality	High	Moderate (less detailed)	Very High
Training Stability	Challenging	Stable	Moderate
Mode Collapse	Yes	No	No
Sampling Time	Fast	Fast	Slow
Latent Space	Unstructured	Structured	Not typically used
Complexity	High (due to the adversarial structure)	Moderate	High

choice for tasks that require interpretability and control over the generation process. However, the trade-off for this stability and control is that VAE-generated samples may lack the detail and complexity of real data, as the model prioritizes efficient data compression over capturing nuance [79,106]. Diffusion models represent a significant advancement in generative modelling capabilities. They achieve state-of-the-art sample quality and diversity while inherently avoiding mode collapse. These attributes make diffusion models a powerful tool for generating highly realistic and diverse data [2,107]. However, a major drawback is their lengthy sampling times [108]. Unlike GANs and VAEs, which can generate an image in a single step, diffusion models require a multi-step denoising process, making them computationally expensive. Additionally, training diffusion models can be complex [109,110]. Table 1 presents a summary of the trade-offs of these generative models.

Despite significant efforts over the past three decades to automate the design of proteins [111,112], the majority of *de novo* designs have yet to achieve the complexity and diversity observed in natural macromolecules. Several factors contribute to this disparity. Modelling the complex relationship between sequence and structure poses a significant challenge [113]. Moreover, most computational design methodologies rely on iterative search and sampling processes, resembling evolution [114]. Although computational techniques have been developed to facilitate searching [112] and accurately predicting natural protein structures [11], effectively exploring the domain of designable protein structures within the vast space of potential proteins remains an ongoing challenge [115]. The objective of protein backbone generation is to create novel structures using information derived from real data distributions. To achieve this objective, one must establish a link between known distributions, such as Gaussian distributions, and the intricate, high-dimensional, and sparse distributions of real data.

Over the past decade, structure-based protein design has emerged as a potent approach to addressing complex challenges, including enzyme catalysis, viral inhibition, and *de novo* structural generation [116–120]. The Rosetta framework [37] has been instrumental in achieving successful protein designs, typically following a two-step process: generating a protein backbone and designing a sequence aimed at minimizing the folded-state energy of the generated backbone. With existing methodologies predominantly limited to the generation of either small molecules [121] or large proteins under highly constrained conditions, often restricted to a single domain topology [122], 3-D molecular structure generation remains a challenge.

In this review, we investigate recent advancements aimed at generating protein 3-D structures and, in certain cases, protein sequences, with a particular focus on diffusion models and equivariant graph neural networks (EGNNs). We explore the mathematics of two diffusion architectures, namely, the denoising diffusion probabilistic models (DDPMs) and the denoising diffusion implicit models (DDIMs), and comprehensively discuss the significance of equivariance in protein generation, highlighting EGNNs as a solution to this challenge.

The remainder of this review is organized as follows. The concept of protein generation is explained in Section 2. Section 3 outlines the diffusion models. Section 4 explains graph neural networks (GNNs) and EGNNs. Related studies addressing the protein generation problem are explored in Section 5. The datasets commonly used in these studies are presented in Section 6. Section 7 concludes the work.

2. Protein representation

A protein is characterized by a distinct linear arrangement of amino acids known as its primary structure, which acts as the blueprint for its folded configuration and ultimately its 3-D structure [123]. These linear chains fold due to intramolecular interactions guided by the sequence. This intricate folding process is primarily driven by the hydrophobic effect, in which hydrophobic amino acid side chains cluster together to minimize unfavourable interactions with the surrounding aqueous environment, ultimately leading to the formation of the protein's unique and functionally relevant 3-D structure [124]. The polypeptide backbone refers to the continuous sequence of atoms that forms the central structural element of the polypeptide chain, as illustrated in Fig. 1 [125]. The global geometric structures of proteins are predominantly determined by their backbones [126]. The protein backbone's specific folding pattern, driven by the minimization of free energy [125], guides the positioning of side chains, contributing to the formation of the protein's binding sites [11], thus enabling it to perform its tasks [125,127]. The extensive array of potential configurations of the peptide backbone, coupled with the desirable chemical bonding geometry and interactions, presents a challenge in protein structure modelling [128].

A protein backbone spans from the N-terminus to the C-terminus with four heavy atoms connected covalently. This arrangement symbolized as N–C_α–C–O, guides the sequential folding from the N-terminus to the C-terminus, with the C_α atom situated centrally. Developing novel protein backbones with desired properties presents a significant challenge due to the inherent complexity of the structure–sequence relationship [90]. The physical and chemical constraints dictate the permissible atomic arrangements within protein structures, while simultaneously demanding "designability" – the existence of an amino acid sequence capable of folding into the specified structure [90].

In the past decade, deep learning has had a substantial impact on protein biology, leading to breakthroughs ranging from sequence embeddings to structure and function prediction. Among these techniques, convolutional neural networks (CNNs) have gained much attention for their ability to extract features directly from protein sequences and structures [17,129–131]. In predicting protein interactions, most sequence-based methods employ 1-D CNNs to capture the sequential patterns in the protein's primary structure [33,132], whereas 3-D CNNs offer similar benefits for structural data, although processing high-resolution representations can be computationally expensive and inefficient as protein structures only occupy a small fraction of the surrounding 3-D space [132].

Unlike CNNs, which are well suited for grid-like data, graph neural networks (GNNs) offer a unique advantage in deep learning, especially for data representations that lack a regular grid structure (i.e. non-Euclidean data structures) [133–136]. GNNs efficiently process information propagating across the complex connections within a graph, making them ideal for analyzing complex networks like protein structures.

These graphs consist of nodes, which represent entities, connected by edges, which represent relationships. This makes GNNs a perfect fit for protein analysis, as protein structures can be naturally represented as graphs, with amino acids modelled as nodes and the bonds between them as edges. GNNs efficiently process information propagating across these complex networks, enabling them to capture the essential geo-

metric relationships between amino acids [137]. This allows GNNs to achieve remarkable results in diverse prediction tasks, ranging from drug activity to protein–protein interaction. By learning hidden patterns in protein structures, GNNs can estimate the conformational energy, predict protein binding, and facilitate drug discovery [138], offering valuable insights into protein stability and folding [139,140].

GNNs handle complex graph structures effectively due to their ability to propagate information across connections [137,141]. Geometric deep learning [142,143], particularly graph convolutional networks (GCNs) [133], circumvents the limitations of explicit 3-D models by generalizing convolutional operations on efficient graph-like molecular representations. They have achieved success in tasks spanning from drug activity [144] to protein interface prediction [145]. To this end, one can represent the 3-D structure of a protein in the form of a graph, using $[C_x, C_y, C_z]$ to denote the 3-D coordinates of the C_α in the backbone and A_i for the ith amino acid among the 20 standard amino acids [9,33]:

$$S = \{([C_x, C_y, C_z], A_i)\} \tag{1}$$

There exists a variety of GNNs including graph convolutional networks (GCNs) [133], graph attention networks (GATs) [146], graph recurrent networks (GRNs) [147], and graph autoencoders [148]. In general, these models extend operations from Euclidean data with grid or sequential structures to graph data. For instance, a node representation is constructed using GCNs by aggregating the features of its neighbors within the graph, effectively expanding the receptive field of the corresponding neuron [136,148]. Within protein structures, GCNs analyze interactions between residues, considering both geometric and biochemical information [145,149,150]. This allows them to achieve success in tasks like protein design [22], function prediction [151], and binding prediction [152].

3. Diffusion models

Diffusion probabilistic models have emerged as pivotal tools in computational biology, specifically in the field of *de novo* protein design and protein sequence and backbone generation, owing to their multifaceted capabilities [2,6,44,153,154]. First, diffusion probabilistic models excel in processing high-dimensional data, which is vital in capturing the complex dependencies intrinsic to protein structures [155]. Second, they adeptly capture the structural variability of proteins, facilitating the generation of a broad spectrum of biologically relevant conformations. Third, their probabilistic framework enables the generation of probability distributions of potential structures, effectively accommodating the inherent uncertainty prevalent in molecular structures [38,89]. These features have led to many innovative applications in protein engineering, drug discovery, and synthetic biology, including accurate structure prediction, *de novo* protein design, and dynamic protein simulation.

This section explores the potential of diffusion models in protein generation, leveraging the representation of proteins as graphs. Additionally, it reviews equivariant graph neural networks (EGNNs), which are crucial for handling dataset transformations and ensuring robustness, interpretability, and generalization across different datasets and transformations. These considerations ultimately lead to improved performance and new insights in protein-related research.

Inspired by principles from non-equilibrium statistical physics [51], diffusion models address a core challenge in machine learning: effectively modelling complex datasets using adaptable probability distributions while maintaining computational tractability. Fundamentally, diffusion models operate by incrementally introducing noise to a dataset through an iterative process known as forward diffusion. Following this phase, a reverse process is employed to reconstruct the original structure of the data, resulting in a flexible and tractable generative model. Expanding on this principle, denoising diffusion probabilistic models (DDPMs) [52] show the potential to achieve performance comparable to or exceeding that of other generative models, such as decoders, energy-based models, and GANs, in image and video generation [75,80,156–158].

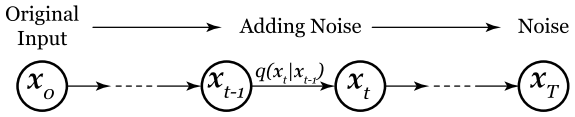

Fig. 2. Forward diffusion process [52]. The initial input is progressively transformed, resulting in a Markov chain that eventually converges to a state of pure Gaussian noise.

Fundamentally, diffusion models involve two processes: the forward process iteratively introduces noise to perturb the original data stepwise from $\boldsymbol{x}_0$ to $\boldsymbol{x}_T$ (Fig. 2), while during the reverse process, a trained neural network recovers the original data from $\boldsymbol{x}_T$ using a denoising process (Fig. 3). This trained diffusion model acts as a generative tool, wherein the generation of data involves passing randomly sampled noise through the learned denoising process. A DDPM is a latent variable model that employs a fixed Markov chain to establish a mapping to the latent space. This Markov chain, comprising latent variables $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$, maintains a temporal dependence solely on the previous time step. Gradually introducing noise to the data within this chain facilitates the approximation of the posterior distribution $q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)$, where $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)$ denote latent variables sharing the same dimensionality as $\boldsymbol{x}_0$ [52].

One may differentiate DDPMs from other latent variable models [159] by their fixed approximate posterior $q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)$, known as the forward diffusion process. This process operates as a Markov chain, gradually introducing Gaussian noise to the data following a specified variance schedule $\beta_1, \ldots, \beta_T$. Moreover, the latent variables in DDPMs have relatively high dimensionality. For instance, Ho et al. [52] examined a Markov chain characterized by Gaussian transitions, with parameters defined by a decreasing sequence $\alpha_{1:T} \in (0,1]^T$ [53]. During the forward process, the transitions within the sampling chain can be configured as conditional Gaussian distributions, especially when the noise amplitude remains sufficiently low. This observation, combined with the Markov assumption, results in a parameterization of the forward process:

$$q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) := \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}) \tag{2}$$

where $\beta_1, \ldots, \beta_T$ denote a variance schedule, which can either be learned or predetermined. This schedule, if appropriately designed, ensures that for sufficiently large T, the latent variable $\boldsymbol{x}_T$ approximates an isotropic Gaussian distribution. The variance schedule dictates the magnitude of the noise introduced at each temporal step. Visualizing it as a map, time serves as the key, while the associated value indicates the degree of noising desired. Using the notation $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$, the following distribution may be used to generate a sample $\boldsymbol{x}_t$ [52]:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) := \mathcal{N}(\boldsymbol{x}_t; \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}) \tag{3}$$

Eq. (3) may be rewritten as $\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$ for $\boldsymbol{\epsilon} \in \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ [52].

As $T \to \infty$, the latent variable $\boldsymbol{x}_T$ approaches an isotropic Gaussian distribution under the diffusion process Eq. (3). This property allows efficient generation of novel data points from the original distribution if the reverse distribution $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ can be learned. By sampling $\boldsymbol{x}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and iteratively applying the reverse process, a sample $\boldsymbol{x}_0$ is obtained, corresponding to a novel data point. However, estimating $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ involves intractable computations requiring the original data distribution [53]. Thus, $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ is approximated by a neural network, p_θ. Since, for small β_t, $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ is approximately Gaussian, p_θ can be chosen to be Gaussian,

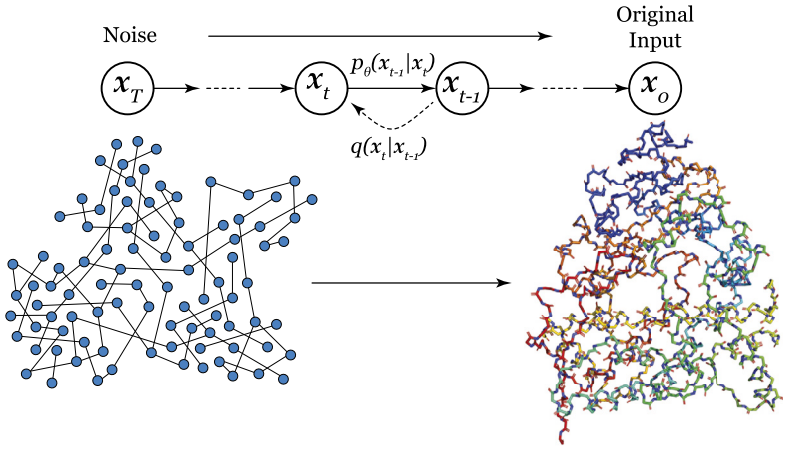

Fig. 3. Reverse diffusion process [52]. The noise component is systematically restored into a sample originating from the target distribution, reversing the transformation. The principal objective in training a diffusion model is to master this reverse process, specifically training $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$. Upon retracing our steps along this sequential progression, novel data can be generated.

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) := \mathcal{N}(\boldsymbol{x}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t, t)) \tag{4}$$

where $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) = \left(\boldsymbol{x}_t - \beta_t \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)/\sqrt{1-\bar{\alpha}_t}\right)/\sqrt{\alpha_t}$. To simulate the reverse process, one must choose the architecture for the neural network $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$. One architecture commonly used in previous studies [52,53] is U-Net [160]. Then, the reverse process (generative process) $\boldsymbol{x}_T \to \boldsymbol{x}_0$ is defined as follows [52]:

$$p_\theta(\boldsymbol{x}_{0:T}) := p(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) \tag{5}$$

where $p(\boldsymbol{x}_T) = \mathcal{N}(\boldsymbol{x}_T; \boldsymbol{0}, \boldsymbol{I})$. During the training process, data points $\boldsymbol{x}_t$ are sampled from a distribution $q(\boldsymbol{x}_t|\boldsymbol{x}_0)$. Then, following the forward process $\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, the aim is to minimize an objective function $T^{-1}\sum_{t=1}^{T} \mathbb{E}_{q(\boldsymbol{x}_0, \boldsymbol{x}_t)}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\|^2]$, which measures the average squared difference between the true noise $\boldsymbol{\epsilon}$ and the noise approximated with a neural network (e.g. U-Net or EGNN), $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$, using stochastic optimization techniques [52]. By simulating the reverse process, one can generate new samples from $p_\theta(\boldsymbol{x}_0)$. This is achieved by initially sampling noise from $\boldsymbol{x}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and then iteratively denoising samples for $t = T-1, \ldots, 0$ as $\boldsymbol{x}_t \sim p_\theta(\boldsymbol{x}_t|\boldsymbol{x}_{t+1})$ [44].

As mentioned in Section 2, given that proteins occupy arbitrary positions in 3-D space, graph structures provide a more effective representation, and thus one can employ a graph-based neural network to learn $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$. For geometric systems such as molecules, in which atomic forces and dipoles are linked to spatial arrangement, it is necessary for models to maintain this relationship under transformations. This property, known as equivariance, ensures that predicted values adapt seamlessly when the molecule's orientation or position changes [161].

Invariance and equivariance are crucial properties for deep learning models, especially when dealing with data that exhibit certain natural or geometric structures, as is the case for a protein backbone. Invariance describes a model's ability to recognize the same object even when certain transformations (e.g. rotation, scaling) are applied to the input. This improves model generalizability to unseen data with slight variations. For instance, an image recognition model should be able to recognize a dog regardless of its pose. Equivariance refers to a property in which the output of a model remains consistent or transforms predictably with respect to transformations applied to the input. This ensures consistency and reduces the need for learning redundant features across different variations.

Achieving invariance and equivariance is a key objective in deep learning, resulting in more interpretable and physically plausible predictions and reducing the parameters to be learned while preserving representational capabilities [162]. Incorporating equivariance principles is vital for successful protein generation as it reduces the protein design space [153]. For example, if a binding site undergoes rotation or translation, a corresponding transformation is expected in the generated molecules [163]. A special case of equivariance is "invariance", in which identical outputs are generated for scalar values like distance or energy [164].

Although graph neural networks (GNNs) are permutation equivariant, they cannot naturally retain symmetries present in 3-D space, so they fail to capture the impacts of invariance and equivariance to rotations, reflections, and translations [165,166]. For instance, given an amino acid in the protein structure shown in Fig. 1, the locations of four backbone atoms (carbon, nitrogen, and oxygen) determine a local skeleton, and different residues interact with each other by performing specific rotations between their local frames, with important impacts on the protein structure and its function [167]. It is, therefore, necessary to incorporate invariance-aware components such as EGNNs.

The expressions E(3), SE(3), O(3), and SO(3) are used to refer to different types of transformations in 3-D space commonly considered in robotics, computer graphics, computational biology, and computer vision. Here, we explain them in detail [168]:

1. O(3) denotes the orthogonal group in three dimensions. It comprises transformations that preserve distance in 3-D space, namely rotations, reflections, and combinations of these operations [169, 170]. They are represented using 3×3 orthogonal matrices:

$$\mathrm{O}(3) = \{\boldsymbol{R} : \mathbb{R}^{3\times 3} | \boldsymbol{R}^T \boldsymbol{R} = \boldsymbol{I}\} \tag{6}$$

2. SO(3) (a subgroup of O(3)) is the special orthogonal group in three dimensions, which represents rotations in 3-D space that preserve distances, angles, and orientation. Unlike O(3), SO(3) transformations do not involve reflections [169,170]. They are characterized by 3×3 orthogonal matrices with a determinant of $+1$:

$$\mathrm{SO}(3) = \{\boldsymbol{R} \in \mathbb{R}^{3\times 3} : \boldsymbol{R}^T \boldsymbol{R} = \boldsymbol{I} = \boldsymbol{R}\boldsymbol{R}^T, \det(\boldsymbol{R}) = 1\} \tag{7}$$

3. E(3) is the group of 3-D Euclidean transformations. These transformations include translation, rotation, and reflection, so E(3) is similar to O(3) but also includes translations [169,170]. They are represented using 4×4 matrices, in which the upper-left 3×3 submatrix represents rotation, and the rightmost column represents translation:

$$\mathrm{E}(3) = \{\boldsymbol{T} \in \mathbb{R}^{4\times 4} | \boldsymbol{T} = \left[\begin{array}{c|c} \boldsymbol{R} & \boldsymbol{t} \\ \hline \boldsymbol{0} & 1 \end{array}\right] : \boldsymbol{R} \in \mathrm{O}(3), \boldsymbol{t} \in \mathbb{R}^3\} \tag{8}$$

4. SE(3) (a subgroup of E(3)) is the special Euclidean group in three dimensions, which includes both translations and rotations (i.e.

SO(3) together with translations), similar to the Euclidean transformations but excluding reflections [169,170]. SE(3) transformations are represented by 4×4 matrices, with the rotation part in the upper-left 3×3 sub-matrix and the translation part in the rightmost column:

$$\mathrm{SE}(3) = \{\boldsymbol{T} \in \mathbb{R}^{4\times4} | \boldsymbol{T} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ 0\ 0\ 0 & 1 \end{bmatrix} : \boldsymbol{R} \in \mathrm{SO}(3), \boldsymbol{t} \in \mathbb{R}^3\} \tag{9}$$

Consider chirality, the ability of a molecule to exist in two asymmetrical forms, known as enantiomers, that are mirror images of each other, with identical atomic composition, connections, and bond orders [171]. One cannot superimpose these enantiomers (identified as left-handed and right-handed), since there always exists at least one substituent linked to the chiral atom that defies superimposition [171]. Due to the chirality of amino acids, proteins, being constructed from these molecules, inherently possess chirality as well [172]. Proteins predominantly exhibit right-handed alpha helices; left-handed helices are not commonly observed in stable forms within natural proteins [44,153]. The network should be capable of discerning these characteristics, requiring a framework that is sensitive to reflection (i.e. that is non-equivariant to reflection) [173].

E(3)-equivariant EGNNs [174] offer invariance to translation, rotation, and reflection in 3-D space. They can scale to higher dimensions and maintain permutation equivariance. Conversely, SE(3)-equivariant models do not commute with input reflections. Therefore, they are particularly well suited for molecular data, for which the chirality of molecules is crucial, especially in proteins. SE(3)-equivariant networks ensure that reflections do not produce biologically inaccurate left-handed structures, thus preserving the inherent chirality of the protein data [153].

Recent years have seen a surge in interest in geometry-aware neural networks specifically designed to generate protein 3-D structures. These studies have explored the power of 3-D rigid transformations, leading to the development of networks that remain unchanged or behave consistently under various transformation groups, including E(3) [44,166,175,176], SE(3) [90], and SO(3) [38,177,178].

Ongoing research endeavors aim to refine the architecture and training methodologies of diffusion models to achieve further performance enhancements. Notably, denoising diffusion implicit models (DDIMs) exhibit enhanced sampling speed, further improving the capabilities of diffusion models. The following section provides additional insights into the architecture of DDIMs, which integrate a non-Markovian inference model.

3.1. Denoising diffusion implicit models (DDIMs)

Denoising diffusion probabilistic models (DDPMs) have shown promise, but their speed is a limiting factor. This section introduces denoising diffusion implicit models (DDIMs), which offer significant improvements. Despite sharing a common objective function, DDIMs outperform DDPMs in sample generation quality, particularly when the sampling is accelerated by 10–100 times [53,179]. DDIMs accelerate the reverse process of diffusion models by employing a non-Markovian sampling process that allows for a less restrictive step schedule than in Markovian DDPMs. Consequently, DDIMs can achieve high-quality sample generation in substantially fewer steps than DDPMs, resulting in a significant computational speed-up. Additionally, unlike DDPMs, DDIM samples exhibit a unique property called “consistency”, meaning they maintain high-level structural and semantic similarity even when generated using different sampling chain lengths [53]. This consistency is evident when generating multiple samples using Markov chains of different lengths, and facilitates semantically meaningful interpolation. Unlike DDPMs, which have inherent randomness, DDIMs allow precise control over the initial state by manipulating the latent variable. This manipulation directly influences the outcome within the model’s latent space, facilitating smooth and controlled transitions between different concepts. This ability for controlled manipulation significantly enhances the expressive power and versatility of DDIMs, making them well suited for various applications requiring controlled manipulation within a defined space.

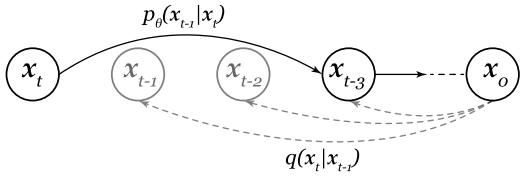

Fig. 4. Accelerated sampling process (DDIM).

The forward diffusion process in DDIMs shares similarities with that used in DDPMs [51,52], with both methods gradually introducing Gaussian noise to the latent variables:

$$q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) := \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) := \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}) \tag{10}$$

The variance schedule in DDIMs, denoted by β_t, can be reformulated using the following notation: $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$. This allows us to rewrite Eq. (10) as follows [52]:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t; \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}) \tag{11}$$

The reverse diffusion process is defined as

$$p_\theta(\boldsymbol{x}_{0:T}) := p_\theta(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) \tag{12}$$

where

$$p_\theta(\boldsymbol{x}_T) = \mathcal{N}(\boldsymbol{x}_T; \boldsymbol{0}, \boldsymbol{I}) \tag{13}$$

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \hat{\boldsymbol{x}_0}) \tag{14}$$

Given $\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)$ and $\epsilon_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, the latent variable $\boldsymbol{x}_t$ can be computed using Eq. (16). The model $\epsilon_\theta^{(t)}(\boldsymbol{x}_t)$ is then employed to predict ϵ_t based solely on the information contained in $\boldsymbol{x}_t$, without any knowledge of $\boldsymbol{x}_0$. Consequently, using Eq. (16), one can predict the denoised observation $\boldsymbol{x}_0$ given $\boldsymbol{x}_t$. This denoising process originates from the inference distribution q_σ and involves substituting the true value of $\boldsymbol{x}_0$ with an estimated value, as described by Zhang et al. [180],

$$\hat{\boldsymbol{x}_0} = f_\theta^{(t)} := \frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}.\epsilon_\theta^{(t)}(\boldsymbol{x}_t)}{\sqrt{\alpha_t}} \tag{15}$$

which is produced by a (reparameterized) neural network that predicts $\boldsymbol{x}_0$ from $\boldsymbol{x}_t$ by minimizing the mean squared error [180]. Eq. (15) provides a way of estimating the final generation as a deterministic function of $\boldsymbol{x}_t$. The latent variable $\boldsymbol{x}_t$ can be defined as

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\boldsymbol{x}_0 + \sqrt{1-\alpha_t}\epsilon \tag{16}$$

As shown in Fig. 4, DDIMs use a non-Markovian inference model for the reverse diffusion process. This process involves an additional inference distribution q_σ, which is designed to have conditional distributions for each latent variable, closely resembling the desired distribution q:

$$q_\sigma(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) = q_\sigma(\boldsymbol{x}_T|\boldsymbol{x}_0) \prod_{T}^{t=2} q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \tag{17}$$

where

$$q_\sigma(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_T; \sqrt{\bar{\alpha}_T}\boldsymbol{x}_0, (1-\bar{\alpha}_T)\boldsymbol{I}) \tag{18}$$

$$q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_{t-1}; \mu_t, \sigma^2 \boldsymbol{I}) \tag{19}$$

and μ_t is defined as

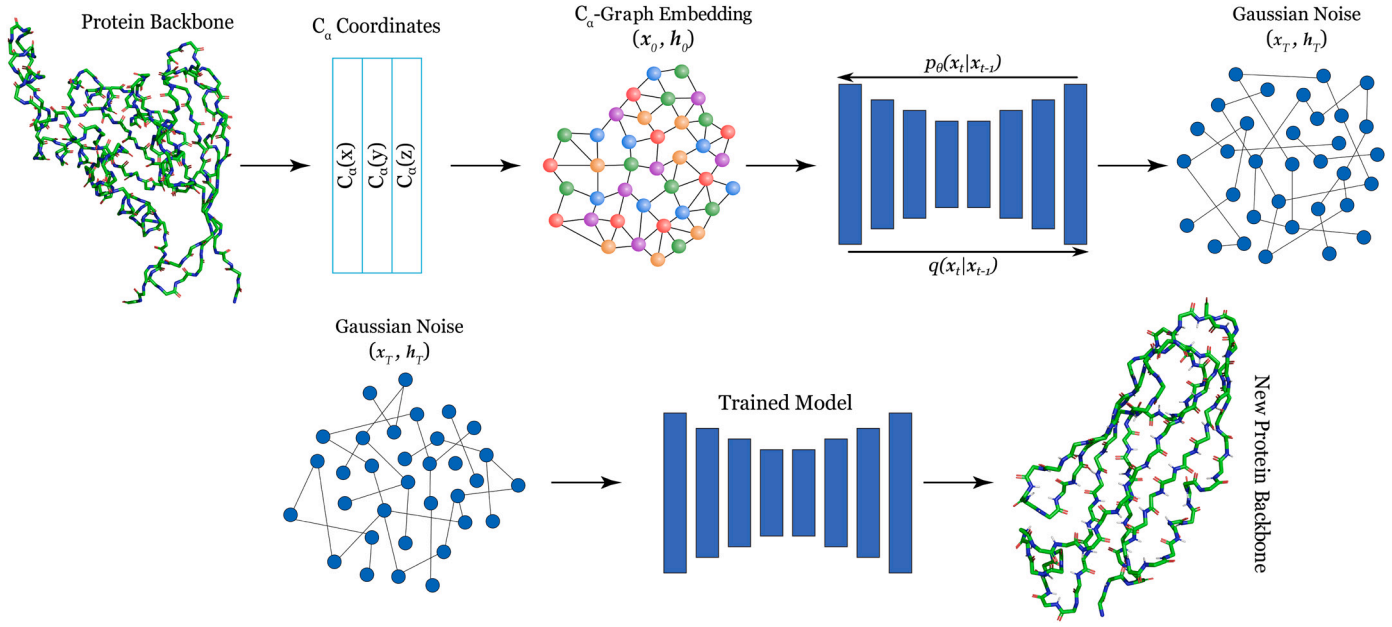

Fig. 5. Protein backbone generation using a diffusion model and graph embeddings.

$$\mu_t = \sqrt{\bar{\alpha}_{t-1}}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\frac{\boldsymbol{x}_t - \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0}{\sqrt{1-\bar{\alpha}_t}} \tag{20}$$

The generative process with fixed prior may be defined as $p_\theta(\boldsymbol{x}_T) = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ and

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \begin{cases} \mathcal{N}(f_\theta^1(\boldsymbol{x}_1, \sigma_1^2\boldsymbol{I}) & \text{if } t=1 \\ q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, f_\theta^t(\boldsymbol{x}_t)) & \text{otherwise,} \end{cases} \tag{21}$$

Following Eq. (21), a sample $\boldsymbol{x}_{t-1}$ can be generated using $\boldsymbol{x}_t$:

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}}\underbrace{\left(\frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}\epsilon_\theta^{(t)}(\boldsymbol{x}_t)}{\sqrt{\alpha_t}}\right)}_{\text{Predicted } \boldsymbol{x}_0} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2}\epsilon_\theta^{(t)}(\boldsymbol{x}_t)}_{\text{Direction pointing to } \boldsymbol{x}_t} + \underbrace{\sigma_t\epsilon_t}_{\text{Random noise}} \tag{22}$$

The initial step, denoted by $\alpha_0 := 1$, corresponds to the initial step in the diffusion process, when no noise has been added, and $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ defines the standard Gaussian noise added at each step, which remains independent of the latent variable $\boldsymbol{x}_t$. The mean of the noisy model at step t is given by $\left(\boldsymbol{x}_t - (1-\alpha_t)\epsilon_\theta(\boldsymbol{x}_t, t)/\sqrt{1-\bar{\alpha}_t}\right)/\sqrt{\alpha_t}$ [181].

The distinct choices of σ values introduce diverse generative processes while using the same model ϵ_θ, negating the need for model re-training. Specifically, when we set $\sigma_t = \sqrt{(1-\alpha_{t-1})/(1-\alpha_t)} \times \sqrt{(1-\alpha_t)/\alpha_{t-1}}$ for all t, the forward process is Markovian, and the generative process aligns with the characteristics of a denoising diffusion probabilistic model (DDPM). Alternatively, in the event that $\sigma_t = 0$ for all t, the forward process becomes deterministic, given $\boldsymbol{x}_{t-1}$ and $\boldsymbol{x}_0$, except at $t = 1$. Consequently, in the generative process, the coefficient preceding the random noise ϵ_t becomes zero. In this case, the resulting model takes the form of an implicit probabilistic model [179], wherein samples are generated from latent variables through a predetermined procedure, spanning from $\boldsymbol{x}_T$ to $\boldsymbol{x}_0$.

To generate instances, one can control the sampling rate τ and σ to interpolate between a deterministic DDIM and a stochastic DDPM [53]:

$$\sigma_{\tau_i}(\eta) = \eta\sqrt{\frac{1-\alpha_{\tau_{i-1}}}{1-\alpha_{\tau_i}}}\sqrt{\frac{1-\alpha_{\tau_i}}{\alpha_{\tau_{i-1}}}} \tag{23}$$

where $\eta \in \mathbb{R}_{\geq 0}$ is a hyperparameter that can be directly controlled. The trainable generative process is denoted by $p_\theta(\boldsymbol{x}_{0:T})$, where each $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ leverages knowledge of $q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$. Given a perturbed observation $\boldsymbol{x}_t$, the preliminary procedure entails constructing a predictive approximation for $\boldsymbol{x}_0$. Subsequently, one may obtain a sample $\boldsymbol{x}_{t-1}$ through the reverse conditional distribution $q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$:

$$q_\sigma(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\left(\sqrt{\alpha_{t-1}}\boldsymbol{x}_0 + \sqrt{1-\alpha_{t-1}-\sigma_t^2}.\frac{\boldsymbol{x}_t - \sqrt{\alpha_t}\boldsymbol{x}_0}{\sqrt{1-\alpha_t}}, \sigma_t^2\boldsymbol{I}\right) \tag{24}$$

4. Equivariant graph neural networks (EGNNs)

In 3-D space, proteins adopt complex and arbitrary structures that can be effectively represented using graphs. However, traditional graph neural networks (GNNs) struggle to maintain geometric symmetries (geometric equivariance). Proteins do not possess an invariant representation due to their arbitrary poses, which include variations in translation and rotation. To apply GNNs to proteins, one must extract translation- and rotation-invariant features, such as the distance matrix. Equivariant graph neural networks (EGNNs) [174] are a type of GNN designed to process such data while maintaining geometric equivariance [177,182]. This is crucial when modelling molecules, and especially proteins, for which outcomes should remain consistent and predictable despite rotations and translations [182]. EGNNs excel at accurately representing the irregular, non-grid structures of proteins, enabling precise modelling of their geometric properties, symmetries, and transformations [183]. In contrast, 3-D convolutional neural networks (3DCNNs) [184] struggle with these complexities due to their reliance on localized convolutions, which prevents them from accurately capturing intramolecular contacts, internal chemistry, and long-range dependencies within protein structures [177,182,185,186]. By integrating EGNNs into diffusion models for protein generation, researchers can effectively capture all pairwise interactions between residues along the protein backbone. Additionally, this approach enables learning of the probability density function score for the denoising process [161,174].

A protein's representation can include the 3-D coordinates of the C_α atoms in the backbone, resembling point clouds. EGNNs exhibit equivariance with respect to rotation and translation of the protein model [187]. Fig. 5 outlines the procedure of embedding a protein backbone

using an EGNN, relying on its C_α coordinates, and subsequently training a diffusion model capable of generating a novel protein backbone.

Equivariance of a function f under a group G is formally expressed as

$$T_g(f(\boldsymbol{x})) = f(S_g(\boldsymbol{x})), \ \ \forall g \in G \tag{25}$$

where S_g and T_g are transformations associated with the group element g. Put simply, transforming $\boldsymbol{x}$ with S_g, and then applying f, yields the same result as applying f directly to $\boldsymbol{x}$ and transforming the result with T_g [174,188]. For a point cloud $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) \in \mathbb{R}^{N\times 3}$ embedded in a 3-D space, a function f is considered equivariant to rotation (and reflection) if, for $\boldsymbol{z} = f(\boldsymbol{x})$, the relationship $\boldsymbol{Rz} = f(\boldsymbol{Rx})$ holds, where $\boldsymbol{Rx} = (\boldsymbol{Rx}_1, \ldots, \boldsymbol{Rx}_N)$ [174,188]. Similarly, f is translation equivariant if

$$\boldsymbol{z} + \boldsymbol{t} = f(\boldsymbol{x} + \boldsymbol{t}) \tag{26}$$

where $\boldsymbol{x} + \boldsymbol{t}$ represents $(\boldsymbol{x}_1 + \boldsymbol{t}, \ldots, \boldsymbol{x}_N + \boldsymbol{t})$.

We define a protein backbone as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The graph consists of nodes representing individual amino acids, indexed by $n = 1, \ldots, N$ for a backbone comprising N amino acids. Each node stores data, denoted by $\mathcal{V} = (\boldsymbol{x}, \boldsymbol{h})$. Here, $\boldsymbol{x}_n \in \mathbb{R}^3$ represents the 3-D coordinates of the C_α atom in the nth amino acid, and $\boldsymbol{h}_n \in \mathbb{R}^m$ represents a set of m features attributed to the nth node. As an example of integrating an EGNN into a diffusion model, Trippe et al. [44] used sinusoidal positional encoding to incorporate both the sequence position and the diffusion time step into the node features $\boldsymbol{h}_n$:

$$\boldsymbol{h}_n(t) = \begin{bmatrix} \varphi(n,1) \\ \vdots \\ \varphi(n,m) \end{bmatrix} + \boldsymbol{H} \begin{bmatrix} \varphi(t,1) \\ \vdots \\ \varphi(t,m) \end{bmatrix} \tag{27}$$

where $\varphi(x,k) = \begin{cases} \sin(x\pi/N^{(2k/m)}), & k \bmod 2 = 0 \\ \cos(x\pi/N^{(2(k-1)/m)}), & k \bmod 2 = 1 \end{cases}$ and $\boldsymbol{H}$ is a random $m \times m$ orthogonal matrix which transforms the temporal encoding to become orthogonal to the positional encoding [44].

Continuing the approach outlined by Trippe et al. [44] in incorporating EGNN within the diffusion model, the interconnections between nodes are defined by edges, with each edge characterized by edge features $\boldsymbol{e}_{nn'} \in \mathbb{R}^m$, containing pairwise relationships. Similarly, Trippe et al. [44] represented edge features using sinusoidal positional encoding features:

$$\boldsymbol{e}_{nn'} = \begin{bmatrix} \varphi(n-n',1) \\ \vdots \\ \varphi(n-n',m) \end{bmatrix} \tag{28}$$

While an E(n) transformation affects $\boldsymbol{x}$, the node features $\boldsymbol{h}$ remain invariant under these transformations [188]. To sum up, E(3) equivariance ensures that a given function $f : \boldsymbol{x}, \boldsymbol{h} \to \boldsymbol{z}_x, \boldsymbol{z}_h$ behaves consistently under transformations involving all orthogonal matrices $\boldsymbol{R}$ and translations $\boldsymbol{t}$ [174,188]:

$$\boldsymbol{Rz}_x + \boldsymbol{t}, \quad \boldsymbol{z}_h = f(\boldsymbol{Rx} + \boldsymbol{t}, \boldsymbol{h}) \tag{29}$$

The construction of the EGNN involves stacking L equivariant graph convolutional layers (EGCLs) [44,84,161]. The first layer receives initial coordinates and features $(\boldsymbol{x}^0, \boldsymbol{h}^0) = (\boldsymbol{x}, \boldsymbol{h})$ as an input. The updates performed in each layer $l = 1, \ldots, L$ are expressed as $(\boldsymbol{x}^l, \boldsymbol{h}^l) = EGCL[\boldsymbol{x}^{l-1}, \boldsymbol{h}^{l-1}]$, where for each node n,

$$\boldsymbol{x}_n^{l+1} = \boldsymbol{x}_n^l + \sum_{n' \neq n} \vec{w}_{nn'} . \phi_x(\boldsymbol{h}_n^l, \boldsymbol{h}_{n'}^l, d_{nn'}, e_{nn'}) \tag{30a}$$

$$\boldsymbol{h}_n^{l+1} = \phi_h(\boldsymbol{h}_n^l, \sum_{n' \neq n} \tilde{e}_{nn'} \boldsymbol{m}_{nn'}) \tag{30b}$$

Here,

$$\vec{w}_{nn'} = \frac{\boldsymbol{x}_n^l - \boldsymbol{x}_{n'}^l}{\sqrt{d_{nn'}} + \gamma} \tag{31a}$$

$$\boldsymbol{m}_{nn'} = \sum_{n' \neq n} \phi_e(\boldsymbol{h}_n^l, \boldsymbol{h}_{n'}^l, d_{nn'}, e_{nn'}) \tag{31b}$$

$$\tilde{e}_{nn'} = \phi_{\text{inf}}(\boldsymbol{m}_{nn'}) \tag{31c}$$

$$d_{nn'} = \|\boldsymbol{x}_n^l - \boldsymbol{x}_{n'}^l\|_2^2 \tag{31d}$$

and $\phi_{\boldsymbol{x}}, \phi_h$, ϕ_e and ϕ_{inf} denote fully connected neural networks, while $\sqrt{d_{nn'}} + \gamma$ normalizes $\boldsymbol{x}_n^l - \boldsymbol{x}_{n'}^l$ with γ serving as a small positive constant introduced to uphold numerical stability [84,188]. Operating as an attention mechanism, $\tilde{e}_{nn'}$ offers a soft estimation of the edges, with $\phi_{\text{inf}} : \mathbb{R}^m \to [0,1]$ [188]. Following L EGCLs, the EGNN yields coordinate and feature embeddings through non-linear transformations, denoted by $\hat{\boldsymbol{x}}, \hat{\boldsymbol{h}} = EGNN[\boldsymbol{x}^0, \boldsymbol{h}^0]$, adhering to the equivariance property as per Eq. (29) [84,174].

Algorithm 1 Equivariant graph neural network (EGNN).

$\mathcal{G} = (\mathcal{V}, \mathcal{E})$ represents the protein backbone.
Input: $(\boldsymbol{x}^0, \boldsymbol{h}^0)$: initial coordinates and features.
Output: updated node embeddings $\hat{\boldsymbol{x}}, \hat{\boldsymbol{h}}$.
Data: $v_n \in \mathcal{V}$: residue (node) n.
Initialization:
fully connected neural networks $\phi_{\boldsymbol{x}}, \phi_h$, ϕ_e and ϕ_{inf}.
For $l = 1, \ldots, L$:

- Calculate the normalized difference of two nodes' coordinates using Eq. (31a).
- Calculate the message passing between nodes using Eq. (31b).
- Obtain a soft estimate of the edges between nodes using Eq. (31c).
- Calculate the distance between two nodes using Eq. (31d).
- Update node coordinates $\boldsymbol{x}_n^l \to \boldsymbol{x}_n^{l+1}$ using Eq. (30a).
- Update node features $\boldsymbol{h}_n^l \to \boldsymbol{h}_n^{l+1}$ using Eq. (30b).

End
Return $\hat{\boldsymbol{x}}, \hat{\boldsymbol{h}} = EGNN[\boldsymbol{x}^0, \boldsymbol{h}^0]$.

Algorithm 1 outlines the architecture of the EGNN, whose output is used to approximate (learn) $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) = \hat{\boldsymbol{x}} - \boldsymbol{x}_t$ [44].

5. A review of generative models in bioinformatics

This section explores recent advances in using equivariant diffusion models to design and generate small molecules, proteins, and protein–ligand interactions. In the past decade, protein structures have been widely studied [189], with the majority of the research focusing on optimization and aiming to enhance the sampling capabilities of Monte Carlo or molecular dynamics methods. These studies frequently strive to expand the structural space under physiological conditions [189]. Addressing this challenging task, most studies leverage prior knowledge, such as physical models or structural data, to guide search algorithms and sample relevant areas within the vast structure space [190–192]. Presumably, for a given application, there is an *a priori* "plausible" protein space that consists of proteins with predefined desired functions. This motivates the development of sampling methods that explore such spaces.

One of the main challenges in 3-D deep learning pertains to the absence of rotational and translational invariance in 3-D coordinates, making generalizable feature learning and generation difficult. This problem has been addressed by Anand and Huang [12] using a generative adversarial network (GAN) [79] that generated 64-residue peptide backbones. Their approach employed a distance matrix representation, effectively preserving crucial invariances in the protein structure. Subsequent steps involved the reconstruction of 3-D coordinates through a convex optimization algorithm [12] and, later, a learned coordinate recovery module [193].

However, GANs are not without their limitations, including the generation of distance matrices that are not Euclidean valid and the degradation of torsion distributions due to the redundancy within the distance matrix representation under reflection [49]. Consequently, attempts at protein structure generation, such as distance-based GANs, often result in unrealistic structures due to chemical inconsistencies [194]. While other methods are capable of generating protein contacts [12,193,195,196], they rely on external tools for the construction or recovery of 3-D coordinates, which may introduce errors. To address these problems, Eguchi et al. [49] proposed a torsion- and distance-aware approach based on a variational autoencoder (VAE) [197], which directly generates 3-D coordinates for full-atom protein backbones, eliminating the need to recover the distance matrix and avoiding the problem of invalid matrices [198].

Traditional approaches to graph generation rely heavily on predefined graph statistics such as degree distribution and clustering coefficients, often requiring engineered features or the learning of specific kernel functions to capture structural information [199]. In contrast, recent progress in deep neural networks (DNNs) has paved the way for advanced deep generative models. Examples include VAEs [197], GANs [200,201], and normalizing flows [202,203], which have significantly enhanced the performance of graph generation for graph-structured data. Deep generative methods such as GraphVAE [197] and MolGAN [200] have shown promise in estimating the graph distribution and generating molecules, respectively. GraphVAE employs two graph neural networks for encoding and decoding, while MolGAN uses a GAN-based framework combined with reinforcement learning for molecular generation. However, despite their success, these methods still have certain limitations. While effective for small graphs, GraphVAE may become computationally expensive for larger ones due to increased memory demands and $O(k^2)$ parameter counts, with only a minor quality impact at higher k values. A critical challenge in the MolGAN [200] approach is mode collapse, owing to the joint training of a GAN and reinforcement learning objectives, which may result in the generated molecules being restricted to a narrow subset.

GANs had long been the dominant approach to generative tasks. However, a new era began with the seminal work by Sohl-Dickstein et al., [51] which introduced the concept of diffusion models. Building on this foundation, Ho et al. [52] proposed the denoising diffusion probabilistic models (DDPMs), showcasing the ability of diffusion models to achieve performance comparable with other state-of-the-art generative models in image generation tasks. The diffusion-based generative models have recently achieved considerable popularity across various domains. This surge in interest may be attributed to their ability to seamlessly learn complex distributions, handle high-dimensional data, and generate highly diverse outputs [80,82,83].

Ever since diffusion models were introduced, the training strategies and structure of the diffusion network have been further enhanced, leading to notably improved performance exceeding that of GANs in the domain of image synthesis [80]. As a result, diffusion models have come to dominate research on generative tasks. Diffusion models have recently been applied to a wide range of bioinformatics challenges, including single-cell gene expression analysis, protein design, drug and small molecule design, and protein–ligand interaction modelling. Notably, these diffusion-based models outperform their predecessors, such as VAEs and GANs, underlining their immense potential in the field of bioinformatics [2]. Consequently, diffusion models have emerged as a pivotal approach in the domain of protein structure generation, with recent studies [38,44,50,86] exploring their potential, inspired by their successful application in image processing [52,54] and small molecule chemistry [84,121,204].

Trippe et al. [44] introduced ProtDiff and SMCDiff. The former is a method that uses alpha carbon (C_α) coordinates for unconditional protein backbone generation. The latter offers conditional sampling, in which scaffolds are sampled conditioned on a given motif. Their approach achieves SE(3) equivariance by employing an EGNN to represent the protein backbone. Another framework called FoldingDiff [50] defines backbones with bond and torsion angles, employing a sequence diffusion model for novel configurations.

Anand and Achim [38] proposed a diffusion model that allows for the generation of both realistic protein sequences and structures across diverse PDB domains. This model is implemented based on the position and orientation of the C_α atoms. To ensure rotation and translation invariance, an invariant point attention (IPA) module [11] is incorporated. In the training phase, the model changes the intermediate structure, the distance matrix of C_α coordinates, to the ground truth structure at $t = 0$ conditioned by block adjacencies. Sequences are generated using masked language modelling. Additionally, joint sequence–structure modelling is achieved by inpainting missing regions of the structure. The performance is evaluated by comparing the complete structure predicted by AlphaFold [11] using the generated sequence.

In contrast, Genie [89] uses an asymmetric representation of the protein residues during the forward and reverse processes. During the forward process, protein residues are represented as point clouds, whereas a cloud of reference frames is employed for the reverse process. This approach offers a simpler and more efficient way of noising while preserving the full expressiveness of the IPA model during sample generation. Crucially, it does so without violating the Gaussian assumption of DDPMs. Lee et al. [86] proposed the diffusion of 2-D pairwise distances and angle matrices for amino acid residues, but this approach requires further optimization through Rosetta minimization [19].

Hoogeboom et al. [84] proposed an unconditional E(3)-equivariant diffusion model (EDM) to generate 3-D molecules. This model leverages a probabilistic analysis to compute likelihood, incorporating both continuous coordinates and categorical features (atom types H, C, N, O, and F and integer-valued atom charges) during the diffusion process, and doing so without relying on a specific atom ordering. Each molecule is represented as a point cloud $\{(\boldsymbol{x}_i, \boldsymbol{h}_i)\}_{i=1,\ldots,M}$, with corresponding coordinate representation $\boldsymbol{x}_i \in \mathbb{R}^n$ and feature vector $\boldsymbol{h}_i \in \mathbb{R}^{nf}$. The equivariant diffusion process is defined for each coordinate $\boldsymbol{x}_i$ and feature $\boldsymbol{h}_i$ by adding noise to the data. The bond types are predicted using the distances between pairs of atoms and the atom types, as detailed by Garcia et al. [188] Following this, the stability of both atoms and molecules is assessed. The EDM outperforms baseline methods, including G-Schnet [205], equivariant normalizing flows (E-NFs) [188], and graph diffusion models (a non-equivariant variation of EDM) [84]. The equivariant diffusion model demonstrates effective scalability and achieves high precision in learning distributions [84].

Building on the E(3)-equivariant diffusion model (EDM) framework [84], Fu et al. [153] proposed a method to address the inherent complexity of diffusion modelling for 3-D protein structures. This complexity arises from the overwhelmingly large space of possible 3-D protein structures [153]. This framework comprises two stages, namely, generating a latent representation for a protein and decoding it. This is achieved by combining a 3-D graph autoencoder and a latent 3-D diffusion model. This framework tackles three challenges: achieving rotation equivariance in the autoencoder design, accurately reconstructing complex connection information in 3-D graphs during decoding, and developing a specialized latent diffusion process for 3-D latent representations of proteins. In this framework [153], the protein undergoes downsampling to a reduced size, followed by reconstructing the original protein through upsampling of the latent graph representation. Due to the sequential arrangement dictated by the link from the N-terminus to the C-terminus in the amino acid chain of a protein [9,33], C_α atoms are positioned in a fixed order [153] (the sequential order is encoded using sinusoidal positional encoding). The fixed order of C_α atoms can be preserved during upsampling, eliminating the need for reconstructing edge connections, unlike in traditional graph autoencoders. The downsampling and upsampling of these sequential data can be performed using a 1-D convolutional neural network (CNN) [153]. However, owing to the 3-D nature of the protein backbone, it is crucial to ensure equivariance during both the downsampling and upsampling stages. A

traditional CNN exhibits equivariance to translations but lacks equivariance to other transformations, such as rotations and reflections [206], so cannot meet this equivariance requirement, but graph neural networks (GNNs) provide a solution to this problem [174].

EquiPPIS [166] is a symmetry-aware protein–protein interaction (PPI) site prediction method that showcases another application of E(3) EGNNs. This framework directly predicts PPI sites on isolated proteins solely from their 3-D structure, eliminating the need for additional or complex information. EquiPPIS facilitates large-scale PPI site prediction and achieves better accuracy, with structural models predicted by AlphaFold 2 as input, than existing methods that use experimental input [166].

While deep generative models have made significant progress in generating protein structures at the primary, secondary, and tertiary levels, concerns regarding the validity of these structures persist [207]. Addressing this challenge, Rahman et al. [175] proposed an E(3)-equivariant neural network encoding-based graph variational autoencoder (EqEN-GVAE), primarily aiming to generate physically realistic tertiary structures by leveraging graph representation learning. Building on the success of equivariant neural networks in generating realistic small molecule structures [84,188], EqEN-GVAE [175] underscores the necessity of imposing specific constraints via the loss function to control key properties of the generated tertiary structure and ensure the network captures them.

Ingraham et al. [87] proposed a generative model for proteins called Chroma, achieving joint sequence–structure likelihood modelling, subquadratic scalability, and diverse conditional sampling without retraining for new target functions. Building on the principles of diffusion models [51,54], Chroma progressively transforms high-dimensional distributions into simpler, reversible distributions using random graph neural networks [208,209]. These networks are equipped to efficiently process the complex geometric relationships within intricate molecular structures. Chroma offers conditional programmable protein structure generation, validated through *in silico* folding and crystallographic experiments. Unlike previous diffusion models, Chroma's generative architecture learns to reverse a correlated noise process, the distance statistics observed in natural proteins, which adhere to well-understood scaling laws in biophysics [87].

Watson et al. [42] proposed a new protein structure generation method called RFdiffusion, leveraging the frame representation developed by RoseTTAFold, which uses both the C_α coordinates and N–C_α–C rigid orientation for each residue [20]. The training dataset is composed of noised structures sampled from the Protein Data Bank (PDB) for up to 200 steps [210]. The noising process involves two steps. First, C_α coordinates are perturbed with 3-D Gaussian noise, for translation. Second, Brownian motion on the rotation matrix manifold is used to randomize the residue orientations [178,211]. During the reverse process (the generative process), RFdiffusion minimizes the mean squared error (MSE) loss between frame predictions and the true protein structure. This loss, averaged across all residues and without alignment, directs denoising trajectories to converge on designable protein backbone structures [42]. The RF structure employs frame-aligned point error (FAPE) as the prediction training loss. While FAPE remains invariant to the global reference frame, the MSE loss lacks such invariance and thus maintains the continuity of the global coordinate frame across different time steps. To generate a protein backbone, RFdiffusion makes a denoised prediction of randomly initialized residue frames and searches a broad spectrum of potential protein structures from these random initial frames. Through iterative refinement, it progressively narrows down potential protein structures, culminating in predictions closely resembling real proteins. The ProteinMPNN network [212] is employed to design sequences encoding these structures, typically generating eight sequences per design, similar to prior studies [44,50]. Exploration of simultaneously designing both structures and sequences within RFdiffusion was not pursued [42].

Despite the promising results observed in the generation of *de novo* protein backbones through diffusion models, there remains a notable absence in the formulation of a well-established methodological framework for diffusion on SE(3). Yim et al. [90] introduced FrameDiff, a novel approach that combines expressive geometric deep learning techniques with diffusion generative modelling, establishing a theoretical basis for SE(3)-invariant diffusion models across multiple frames. When applied to monomer backbone generation, FrameDiff can generate designable monomers spanning up to 500 amino acids, notably without dependence on a pre-trained protein structure prediction network, unlike earlier methodologies [90].

We highlight that RFdiffusion [42] and Chroma [87] have been experimentally validated [2], unlike the other methods discussed in this section.

Table 2 provides an overview of protein generation methods that employ generative and graph-based methods, and Table 3 lists their contributions and limitations.

Although computational methods like diffusion models and EGNNs have shown remarkable success, it is essential to acknowledge their limitations. Diffusion models require extensive, diverse, and high-quality datasets for effective training [109]. This poses a challenge due to the limited availability of experimental data on protein structures [132,219]. Moreover, achieving optimal performance with either EGNNs or diffusion models requires tuning numerous hyperparameters, a process that can be computationally intensive and often demands extensive experimentation. Furthermore, these methods are computationally expensive to train, particularly for large datasets, and suffer from slow sampling speeds (resulting in low inference speeds) [109,153]. Therefore, they require substantial computational resources, such as powerful GPUs or AI accelerators, which may not be readily available to all researchers. In addition, understanding the underlying mechanisms by which EGNNs and diffusion models generate protein structures is crucial for ensuring the reliability and interpretability of their results. However, achieving robust generalization across diverse protein structures and conditions remains challenging, especially in the absence of clear interpretability [220,221].

6. Datasets

Within computational biology and structural bioinformatics, a range of computational techniques are employed, often relying on input data sourced from either public or private databases (e.g. protein structures and sequences). This section provides an overview of the commonly employed datasets, which are listed in Table 2, including QM9, CATH, GEOM, and PDB.

The QM9 dataset [222] serves as a benchmark database for small molecules, containing 3-D structures, molecular properties, and atom coordinates for 134,000 small molecules (from the chemical universe GDB-17 database [223]) with at most 9 heavy atoms (or 29 atoms including hydrogen). The QM9 dataset includes small amino acids, such as Gly and Ala, as well as nucleobases like cytosine, uracil, and thymine. Calculations have been performed for all 134,000 molecules to determine various properties including equilibrium geometries, frontier orbital eigenvalues, dipole moments, harmonic frequencies, polarizabilities, and thermochemical energetics corresponding to atomization energies, enthalpies, and entropies at ambient temperature [222].

Established in 1997, the Class, Architecture, Topology, and Homology (CATH) database [224] is an up-to-date and systematic classification of protein 3-D structures. The CATH database systematically identifies domains within protein structures sourced from wwPDB and categorizes them into evolutionary superfamilies [225–227], offering both structural and functional annotations. Using a semi-automated approach, CATH splits 3-D structures into their constituent domains, which are semi-independently folding globular units. These domains are then clustered into homologous superfamilies based on discernible evidence of evolutionary ancestry. CATH operates at two levels: CATH-B provides a daily update on domain structures and superfamily assignment. CATH+ includes additional derived data such as predicted

Table 2
Descriptions of the diffusion-based generative methods addressing protein and small molecule generation.

Application	Framework	Year	Generative Model	Method Description
Protein design and generation	Anand and Achim [38]	2022	DDPM	This is an equivariant diffusion model that uses denoising diffusion probabilistic models (DDPMs) and invariant point attention (IPA) modules [11].
	SMCDiff/ProtDiff [44]		DDPM + EGNN	ProtDiff performs E(3)-equivariant unconditional protein backbone generation using C_α coordinates and embedded features. SMCDiff samples scaffolds conditioned on a given motif.
	ProteinSGM [86]		Continuous-time stochastic differential equations (SDEs)	Based on a score-based generative model, ProteinSGM unconditionally generates novel protein structures. It employs continuous-time SDE to translate protein information into four matrices corresponding to beta-carbon pairwise distances, ω and θ torsional angles, and ϕ planar angles. These constraints are then used by RoseTTAFold to build protein structures.
	FoldingDiff [6]		DDPM + vanilla transformer	This framework is a torsion angle-based protein backbone diffusion model. The protein backbone (N–C_α–C) is represented using a sequence of angles (invariant to translation and rotation) that capture the relative orientations of individual amino acid residues.
	DiffSDS [88]	2023	Language diffusion model	Taking inspiration from FoldingDiff [6], DiffSDS derives a 1-D directional representation from invariant atom features, akin to that of the torsion angle representation. This representation allows an encoder–decoder language model to perform the diffusion process. The encoder converts atom features into a hidden atomic direction space, in which equivalent direction vectors reside. The decoder reverses the transformation.
	Chroma [87]		SDEs + GNN	Chroma is a graph neural network (GNN)-based conditional diffusion model that generates large single-chain proteins, and protein complexes exceeding 3,000 residues, with desired properties and functions.
	FrameDiff [90]		SDEs + SE(3) transformer	FrameDiff is an SE(3) diffusion model that uses denoising score matching loss for protein generation. This method achieves the generation of designable monomers up to 500 amino acids in length without the need for a pre-trained protein structure prediction network.
	RFDiffusion [42]		DDPM	RFdiffusion uses RoseTTAFold [20] as the denoising network. It achieves unconditional and topology-constrained protein monomer design up to 600 residues in length, protein binder design, and enzyme active site scaffolding. Owing to the equivariance properties of RoseTTAFold, RFdiffusion maintains rotational symmetry during the prediction.
	LatentDiff [153]		3-D graph autoencoder + DDPM	LatentDiff uses an equivariant protein autoencoder that embeds proteins into a latent space, followed by an equivariant diffusion model that learns the distribution of the latent protein representations.
	Genie [89]		DDPM + SE(3)-equivariant denoiser	This framework performs SE(3)-equivariant discrete-time diffusion to generate protein structures. Similar to the framework introduced by Anand and Achim [38], Genie integrates IPA modules [11] and DDPMs to achieve SE(3) equivariance.
Protein–ligand (small molecules) interaction modelling	DiffSBDD [98]	2022	DDPM + EGNN	This work presents a novel approach to structure-based drug design using an SE(3)-equivariant 3-D diffusion model conditioned on protein pockets to generate new drug ligands. To validate the model performance under realistic binding scenarios, DiffSBDD uses experimentally determined binding data.
Small molecule generation and drug design	EDM [84]		DDPM + EGNN	This method comprises a 3-D graph autoencoder with a latent 3-D diffusion model to generate small molecules. Similarly to ProtDiff, this method uses a densely connected EGNN. Through a probabilistic analysis, the EDM calculates the likelihood by considering both continuous coordinates and categorical features, such as atom types (H, C, N, O, F) and integer-valued atom charges. This computation occurs during the diffusion process, and notably, it does not depend on any specific atom ordering.
	DiffBridge [92]		Deep diffusion generative models + GNN	This model, based on Lyapunov functions, introduces “physically informed” diffusion bridges a novel type of stochastic process”, which guarantees the generation of a specific observation at a fixed time. This allows for training diffusion-based generative models by incorporating physical and statistical prior information.
	SDEGen [94]	2023	SDEs + GNN	A conformation generation model based on stochastic differential equations (SDEs) and generative modelling [54]. Beyond finding single low-energy conformations, this model finds a spectrum of locally optimal conformations.
	GEOLDM [161]		DDPM + EGNN	A novel latent diffusion model that generates 3-D molecules. This model comprises autoencoders that encode structures into continuous latent codes, followed by diffusion models operating in the continuous, lower-dimensional latent space.

Table 3
Contributions and limitations of diffusion-based generative methods addressing protein and small molecule generation.

Application	Framework	Contribution	Limitations
Protein design and generation	Anand and Achim (2022) [38]	This framework generates plausible and realistic protein structures and sequences, performing joint modelling of protein sequences and structures.	The model was trained and tested on a relatively small training dataset. The proposed conditioning (block adjacencies) over generation is limited.
	SMCDiff/ ProtDiff (2022) [44]	SMCDiff was the first to address the motif-scaffolding generation problem using a diffusion model.	This method was trained based on proteins with up to 128 residues from PDB. The absence of torsion angles in the feature representation impedes the model's ability to accurately encode protein chirality, potentially leading to structures with incorrect handedness, such as left-handed helices.
	ProteinSGM (2022) [86]	ProteinSGM offers scaffold inpainting and functional site inpainting, and enables the generation of realistic protein structures that adhere to user-defined constraints.	ProteinSGM relies on post-processing by Rosetta using a Markov chain Monte Carlo method, leading to computationally intensive predictions.
	FoldingDiff (2022) [6]	This framework unconditionally generates diverse and realistic protein structures using a simple transformer model and a DDPM.	Predicting angles sequentially with a transformer model has limitations. Accumulating errors from early predictions can significantly impact the final structure, potentially leading to atom collisions. This method is limited to creating single-chain proteins and the published weights are limited to generating proteins up to length 128.
	DiffSDS (2023) [88]	Geometric constraints and conditional diffusion enable DiffSDS to outperform approaches like RFDesign [213] in restoring protein backbone structures. Additionally, DiffSDS obtains lower connectivity error than RFDesign [213] and FoldingDiff [6]. This indicates DiffSDS's superior ability to effectively connect masked endpoints.	Imposing geometric constraints on the model is explicitly required. Otherwise, DiffSDS cannot generate geometrically sound structures.
	Chroma (2023) [87]	Chroma surpasses the protein size limits for methods such as ProteinSGM [86], Foldingdiff [6], DiffSDS [88], and ProtDiff [44]. The short- and long-range interactions are captured using a random graph neural network, inspired by fast N-body methods [214], leading to sub-quadratic computational scaling. Chroma offers programmable generation of proteins according to user-specified properties, such as residue–residue distances, symmetry, and shape.	This method modifies the diffusion process to improve the accuracy of sampled (at low-temperature) molecular backbones at the expense of reducing the diversity of possible conformations.
	FrameDiff (2023) [90]	This framework has been reported to outperform FoldingDiff [6] and achieve comparable performance with Chroma [87] and RFdiffusion [42], which has four times more parameters.	The study primarily focused on the generation of monomeric proteins up to 500 in length. Evaluation of the method's applicability to multimeric protein design was not addressed by the work.
	RFDiffusion (2023) [42]	RFdiffusion incorporates a self-conditioning prediction strategy, following the success of the recycling process in AlphaFold 2 [11]. In this approach, the current prediction is conditioned on the preceding time step's prediction, leading to a significant improvement in model performance. RFdiffusion conditionally constructs scaffolds for functional motifs and enzyme active sites, using protein motif coordinates as input.	The production of plausible structures for large proteins is challenging to validate computationally due to likely exceeding the capabilities of single-sequence prediction models such as AF2 [11] and ESMFold [215].
	LatentDiff (2023) [153]	LatentDiff reduces the modelling space of protein structures, using a pre-trained equivariant 3-D autoencoder, subsequently transforming protein backbones into a more compact latent space, and models the latent distribution with an equivariant latent diffusion model.	FrameDiff [90] and RFdiffusion [42] generate full backbone atoms, whereas ProtDiff [44] and LatentDiff [153] generate C_α atoms [153].
	Genie (2023) [89]	This framework uses a dual representation for protein residues. During the forward process, proteins are treated as sequences of C_α coordinates. Thus, isotropic Gaussian noise is injected into these coordinates, avoiding the more challenging task of manipulating rotation matrices. Conversely, for noise prediction, protein structures are represented as sequences of Frenet–Serret (FS) frames [216]. This captures crucial information about inter-residue orientations, enabling Genie to generate high-quality protein structures.	The model size is relatively small. This model does not support joint sequence and structure generation. Genie does not offer conditional structure generation.
Protein–ligand (small molecules) interaction modelling	DiffSBDD (2022) [98]	DiffSBDD optimizes arbitrary molecular properties using a noise/denoise scheme and an evolutionary algorithm and addresses molecular design problems using an inpainting-inspired approach. These problems include scaffold hopping/elaboration and fragment growing/merging. DiffSBDD offers *de novo* molecule generation based on protein-conditioned generation and learning the joint distribution of protein and ligand atoms.	The reported performance of this method is lower than some other methods such as Pocket2Mol [217] and GraphBP [163].

Table 3 (*continued*)

Application	Framework	Contribution	Limitations
Small molecule generation and drug design	EDM (2022) [84]	This method directly generates molecules in 3-D space. Unlike autoregressive models, EDM does not rely on a particular atom ordering, offering greater flexibility and substantially more efficient training than normalizing flows. Additionally, EDMs generate up to 16 times more stable molecules than E-NFs [188] when trained on QM9, while requiring half of the training time. Compared with E-NFs [188], EDMs trained on QM9 generate up to 16 times more stable molecules in half the training time. This allows for scaling the training to larger drug-like datasets such as GEOM-Drugs. Unlike previous non-autoregressive models, EDMs accommodate explicit modelling of hydrogen atoms within the generated valid conformations.	In contrast to ProtDiff, EDM does not incorporate sequence-distance edge features for EGNN implementation. Consequently, it does not enforce a sequence order, so lacks a mechanism to establish a relationship between generated coordinates and a backbone chain.
	DiffBridge (2022) [92]	This method achieves high-quality and stable molecule generation and uniformity-promoted point cloud generation.	Energy contributions from the torsional angle are not considered. Deep diffusion bridge models suffer from lengthy training times. Attempts to expedite training with larger batches lead to decreased performance.
	SDEGen (2023) [94]	This model quickly identifies low-energy conformations owing to its ability to capture a multimodal conformation distribution. Additionally, this model demonstrates enhanced conformation generation efficiency, outperforming established approaches such as ConfGF [218]. SDEGen also facilitates interpreting molecular evolution within stochastic dynamic systems.	This model does not cover the crystal conformation RMSD threshold of 1.5 Å. The model performance declines for larger numbers of rotatable bonds, particularly for molecules with eight or more, like those in GEOM-Drugs.
	GEOLDM (2023) [161]	The model achieves rotational and translational equivariance through a point-structured latent space with both invariant scalars and equivariant tensors.	This model has not been evaluated for generating challenging 3-D geometries such as proteins.

sequence domains and functionally coherent sequence subsets known as functional families. The latest release, CATH+ version 4.3, has significantly expanded coverage of both structural and sequence data. It includes 500,238 structural domains and 151 million predicted sequence domains assigned to 5,481 superfamilies.

The Geometric Ensemble of Molecules (GEOM) is a comprehensive dataset featuring high-quality conformers for 317,928 mid-sized organic molecules sourced from experimental data, alongside 133,258 molecules from the QM9 dataset [222]. Additionally, it incorporates data on 304,466 drug-like species and their corresponding biological assay results, obtained from the AICures platform, a collaborative machine-learning initiative aimed at predicting potential drug candidates for the treatment of COVID-19 and related diseases. The dataset also contains 16,865 molecules from the MoleculeNet benchmark [228], each labelled with experimental properties pertaining to physical chemistry, biophysics, and physiology. Conformers within the GEOM dataset were generated using the CREST program [229], which employs extensive sampling based on the semi-empirical extended tight-binding method (GFN2-xTB [230]) to ensure the generation of reliable and accurate structures. Furthermore, conformer ensembles for 1,511 species from the BACE dataset [231] were labelled with high-accuracy single-point DFT energies and semi-empirical quasi-harmonic free energies. Among these ensembles, 534 underwent further refinement through DFT geometry optimizations. GEOM serves to address two prominent gaps in the existing dataset literature. First, it facilitates the benchmarking of new models that use conformers as input to predict various experimental properties, such as biological assay outcomes for antiviral activity or physicochemical and physiological attributes. This fills a void left by previous molecular datasets, which primarily consisted of 2-D graphs or single 3-D structures. Second, GEOM provides an invaluable resource for training generative models aimed at predicting conformers based on molecular graphs, a burgeoning research area seeking to outperform the computational efficiency and accuracy of conventional methods. The dataset's extensive size, simulation accuracy, and connection with experimental data make it an ideal candidate for training generalizable models and pre-training generative models for conformer prediction. Additionally, machine learning models trained on GEOM are anticipated to deliver notable advancements in terms of speed, reliability, and cost-effectiveness compared with traditional conformer generation methods. Despite its strengths, it is important to note that the statistical weights assigned to each conformer by the CREST program may exhibit inaccuracies. Therefore, benchmarks involving conformer probabilities should use the DFT weights provided within the GEOM dataset for enhanced precision and reliability.

The Protein Data Bank (PDB) is recognized as one of the most widely used open-access biodata resources globally. With approximately 450 other database resources downloading, integrating, and distributing PDB data, nearly 200,000 rigorously validated 3-D structures of biological macromolecules, including proteins, nucleic acids, carbohydrates, and their complexes with small molecule ligands, are consistently archived [232–234]. The availability of PDB data without usage restrictions has fostered the development of structural bioinformatics as a vibrant sub-discipline of computational biology. The emergence of artificial intelligence/machine learning methods, such as AlphaFold 2 [11] and RoseTTAFold [20], has revolutionized structural bioinformatics, enabling accurate prediction of protein 3-D structures comparable to low-resolution experimental methods. The distribution of computed structure models (CSMs) currently integrated within PDB (as reported by Burley et al. [234]) is sourced from the following datasets:

- The AlphaFold database [235], generated by AlphaFold 2. [11]
  - Model organism proteomes: 326,175 protein structures representing 48 different model organisms.
  - Global health proteomes: 238,274 protein structures from various disease-causing organisms.
  - Swiss-Prot sequences [236]: 542,380 protein structures, with 430,961 additional structures beyond those in the first two sets.
  - Matched Annotation from NCBI and EMBL-EBI (MANE) sequences [237]: 17,334 protein structures, including 3,844 additional structures not present in the above three sets.
- ModelArchive, using both RoseTTAFold [20] and AlphaFold 2 [11].
  - Core eukaryotic protein complexes: 1,106 models developed by the Baker lab [238].

Table 4 describes the datasets used for training, testing, and validating the methods presented in Table 2.

Table 4
A compilation of the datasets employed for the methods described in this review, along with links to their corresponding open-source code.

Framework	Dataset	Link
ProtDiff [44]	The training data for ProtDiff were restricted to single-chain proteins (monomers) from PDB [210] and lengths between 40 and 128 amino acids.	Code
EDM [84]	The EDM framework was trained on the standard QM9 dataset [222]. The dataset was partitioned into training, validation, and test sets, as outlined by Anderson et al. [239], with 100,000, 18,000, and 13,000 samples, respectively, for each partition.	Code
DiffSBDD [98]	DiffSBDD was trained on the CrossDocked dataset with 100,000 high-quality protein–ligand pairs for the training set and 100 proteins for the test set.	Code
Anand and Achim [38]	This framework was trained on X-ray crystal structure data of CATH 4.2 S95 domains [240,241] from PDB [210]. The training (~ 95%, 13,744 classes, 53,414 domains) and testing (5%, 78 classes, 4,372 domains) datasets were obtained leveraging CATH topology classes. This improves the generalizability of the evaluation by removing sequence and structural redundancy between the datasets.	Code
ProteinSGM [86]	To minimize redundancy and potential bias in specific folds, a subset of the CATH 4.3 dataset [224] containing protein chains with at least 95% sequence similarity was used. From the initial 48,949 unique chains, two filters were applied: 1) A sequence length between 40 and 128 residues. 2) The presence of all backbone heavy atom coordinates N–C_α–C. This resulted in a final set of 10,361 structures, each representing one or more CATH-classified folds.	Code
FoldingDiff [6]	The model was trained using the CATH dataset [242], with chains below 40% sequence identity and less than 60% overlap included. The chains shorter than 40 residues were excluded while chains longer than 128 residues were cropped to a 128-residue window. A standardized 80/10/10 split separated the dataset into 24,316 training backbones, 3,039 validation backbones, and 3,040 test backbones for accurate evaluation.	Code
FrameDiff [90]	The performance of FrameDiff was assessed on monomer backbone generation and trained with $L = 4$ layers on a filtered set of 20,312 backbones taken from the Protein Data Bank (PDB). The evaluation was performed on proteins up to a length of 500.	Code
Chroma [87]	The training datasets were acquired using data from the Protein Data Bank (PDB) as of March 20, 2022, UniProt 2022_01, and PFAM 35. Chroma's training, test, and validation sets are available at Dataset.	Code
RFDiffusion [42]	The training data comprise four datasets: 1) Monomer and homo-oligomer structures sourced from PDB. 2) Hetero-oligomer structures from the PDB, with data collected up to August 2, 2021. 3) Structural models generated by AlphaFold 2, specifically those with a predicted local distance difference test (pLDDT) score exceeding 0.758. 4) Negative protein–protein interaction instances, artificially generated through random pairing. Throughout the training process, examples were sampled from each database, maintaining a ratio of 2:1:4:1.	Code
LatentDiff [153]	LatentDiff was trained on Protein Data Bank (PDB) and Swiss-Prot data in the AlphaFold Protein Structure Database (AlphaFold DB). The single-chain protein data from PDB were restricted to those with C_α–C_α distance less than 5 Å and sequence length between 40 and 128 residues, resulting in 4,460 protein sequences.	Code
SDEGen [94]	This framework was evaluated based on three datasets, GEOM-QM9, GEOM-Drugs [243], and ISO17 [244,94]. GEOM-QM9 is a valuable resource for studying small molecules. This dataset focuses on neutral molecules with a maximum of nine non-hydrogen atoms. The included conformers were generated using the CREST program [229]. Based on the sampling scheme outlined by Shi et al. [218], the dataset was divided into a training set of 40,000 molecules with 200,000 conformations and a test set of 200 molecules with varying numbers of conformations depending on the specific subset (GEOM-QM9: 22,408 conformations; GEOM-Dugs: 14,324 conformations).	Code
DiffSDS [88]	The model was trained using CATH 4.3 [224]. To ensure the test set remained unseen by the model and prevent potential information leakage, proteins with high structural similarity to the training data were filtered. Proteins with a TM-score exceeding 0.5 were excluded from the test set.	Code
Genie [89]	Two publicly available protein structure databases, SCOPe [245,246] and AlphaFold Swiss-Prot [11,235], were used for both training and assessing the framework. The training set contains 8,766 domains, including 3,942 domains with at most 128 residues and 7,249 with at most 256 residues. From AlphaFold Swiss-Prot, only high-confidence protein structures (pLDDT above 80) were included. This resulted in a training set of 195,214 structures, each containing a maximum of 256 residues.	Code
GEOLDM [161]	This model was trained based on the QM9 dataset [222] for both unconditional and conditional molecule generation. The training, validation, and test datasets have 100,000, 18,000, and 13,000 samples, respectively. Additionally, GEOLDM was tested on the GEOM-Drugs dataset [243]. This dataset offers extensive data on large organic molecules (up to 181 atoms, average 44.2, 5 atom types). It includes 37 million conformations for ~450,000 molecules, labelled with energy and statistical weight. Following Hoogeboom et al. [84], the 30 lowest-energy conformations of each molecule were selected for training.	Code

7. Conclusion

This review aims to summarize the methods employed in protein and small molecule generation, focusing on diffusion models and equivariant graph neural networks and their applications in both creating new protein sequences and structures and refining existing approaches. Also explored are the datasets commonly used to train and evaluate these methods.

In the last decade, the rise of generative models and the availability of high-performance computing, thanks to powerful GPUs, have helped solve complex problems such as generating *de novo* proteins and novel drugs [247]. Generative adversarial networks (GANs) [79] have historically held a prominent position in generative tasks. However, a significant shift followed the introduction of diffusion models by Sohl-Dickstein et al. [51] This seminal work paved the way for further advancements, like the denoising diffusion probabilistic models (DDPMs) proposed by Ho et al. [52], demonstrating the potential of diffusion models to challenge state-of-the-art image generation techniques. Several factors explain their surging popularity, including their ability to effectively learn complex distributions, handle high-dimensional data, and generate remarkably diverse outputs [80,82,83]. Following this initial breakthrough, diffusion models have undergone significant

refinements in training strategies and network structures, leading to performance surpassing even the most advanced GANs [80]. This has propelled them to the forefront of protein structure generation, with numerous recent studies [38,44,50,86] exploring their potential.

Emerging research in protein generation builds on the foundation laid by Anand and Achim [38], who established the potential of equivariant diffusion models to generate protein sequences and structures jointly. This was followed by Trippe et al. [44] proposing SMCDiff and ProtDiff methods, the first approaches to addressing the motif-scaffolding generation problem using diffusion models. Additionally, advancements in protein structure prediction tools like RoseTTAFold [20] have inspired approaches such as RFDiffusion [42] and ProteinSGM [86] to achieve protein design for monomers exceeding 600 residues. Notably, Chroma [87] uses a combination of GNNs and diffusion models to generate even larger protein complexes, surpassing 3,000 residues. Building on earlier successes in small molecule and protein generation, recent methods (Tables 2 and 3) focus on creating longer and experimentally validated protein structures and sequences.

Despite the promising results demonstrated by diffusion models, such as their proficiency in binder design and their ability to facilitate the targeting and modulation of diverse protein states [248,249], current methods still face a variety of challenges. These include generating longer protein sequences, managing computational complexity, ensuring experimental validation of generated structures, avoiding unrealistic protein structures, and designing multi-state enzymes. Although experimental findings provide valuable insights into specific signalling proteins and pathways, the features and design principles of multi-state enzymes remain elusive [250]. The modelling of signalling pathways with multi-state enzymes is particularly hindered by the combinatorial explosion of interactions within the system [250]. Tackling these complexities requires further research and innovation in generative methods. For example, ProtDiff [44] is susceptible to generating left-handed helices due to its inherent reflection symmetry, which is not found in stable forms within naturally occurring proteins. Therefore, future efforts may focus on developing a framework capable of generating diverse, novel, and high-fidelity proteins.

CRediT authorship contribution statement

Farzan Soleymani: Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Formal analysis, Data curation, Conceptualization, Investigation, Resources. **Eric Paquet:** Writing – review & editing, Validation, Supervision, Resources, Investigation, Funding acquisition, Formal analysis. **Herna Lydia Viktor:** Supervision, Funding acquisition. **Wojtek Michalowski:** Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Garrett RH. Biochemistry. Cengage; 2015.
[2] Guo Z, Liu J, Wang Y, Chen M, Wang D, Xu D, et al. Diffusion models in bioinformatics and computational biology. Nat Rev Bioeng 2024;2(2):136–54.
[3] Paquet E, Soleymani F, St-Pierre-Lemieux G, Viktor HL, Michalowski W. Quantumbound–interactive protein generation with one-shot learning and hybrid quantum neural networks. Artif Intell Chem 2024;2(1):100030.
[4] Holm L, Sander C. Database algorithm for generating protein backbone and side-chain co-ordinates from a cα trace: application to model building and detection of co-ordinate errors. J Mol Biol 1991;218(1):183–94.
[5] Schenkelberg CD, Bystroff C. Protein backbone ensemble generation explores the local structural space of unseen natural homologs. Bioinformatics 2016;32(10):1454–61.
[6] Wu KE, Yang KK, van den Berg R, Alamdari S, Zou JY, Lu AX, et al. Protein structure generation via folding diffusion. Nat Commun 2024;15(1):1059.
[7] Diem MD, Hyun L, Yi F, Hippensteel R, Kuhar E, Lowenstein C, et al. Selection of high-affinity centyrin fn3 domains from a simple library diversified at a combination of strand and loop positions. Protein Eng Des Sel 2014;27(10):419–29.
[8] Golinski AW, Mischler KM, Laxminarayan S, Neurock NL, Fossing M, Pichman H, et al. High-throughput developability assays enable library-scale identification of producible protein scaffold variants. Proc Natl Acad Sci 2021;118(23):e2026658118.
[9] Soleymani F, Paquet E, Viktor H, Michalowski W, Spinello D. Protein–protein interaction prediction with deep learning: a comprehensive review. Comput Struct Biotechnol J 2022;20:5316–41.
[10] Costantini S, Colonna G, Facchiano AM. Amino acid propensities for secondary structures are influenced by the protein structural class. Biochem Biophys Res Commun 2006;342(2):441–51.
[11] Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, et al. Highly accurate protein structure prediction with alphafold. Nature 2021;596(7873):583–9.
[12] Anand N, Huang P. Generative modeling for protein structures. In: Advances in neural information processing systems, vol. 31. 2018. p. 7494–505.
[13] Mataeimoghadam F, Newton MH, Dehzangi A, Karim A, Jayaram B, Ranganathan S, et al. Enhancing protein backbone angle prediction by using simpler models of deep neural networks. Sci Rep 2020;10(1):19430.
[14] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521(7553):436–44.
[15] Kamisetty H, Ovchinnikov S, Baker D. Assessing the utility of coevolution-based residue–residue contact predictions in a sequence- and structure-rich era. Proc Natl Acad Sci 2013;110(39):15674–9.
[16] Hopf TA, Schärfe CP, Rodrigues JP, Green AG, Kohlbacher O, Sander C, et al. Sequence co-evolution gives 3d contacts and structures of protein complexes. eLife 2014;3:e03430.
[17] AlQuraishi M. End-to-end differentiable learning of protein structure. Cell Syst 2019;8(4):292–301.
[18] Senior AW, Evans R, Jumper J, Kirkpatrick J, Sifre L, Green T, et al. Improved protein structure prediction using potentials from deep learning. Nature 2020;577(7792):706–10.
[19] Yang J, Anishchenko I, Park H, Peng Z, Ovchinnikov S, Baker D. Improved protein structure prediction using predicted interresidue orientations. Proc Natl Acad Sci 2020;117(3):1496–503.
[20] Baek M, DiMaio F, Anishchenko I, Dauparas J, Ovchinnikov S, Lee GR, et al. Accurate prediction of protein structures and interactions using a three-track neural network. Science 2021;373(6557):871–6.
[21] Baek M, Baker D. Deep learning and protein structure modeling. Nat Methods 2022;19(1):13–4.
[22] Strokach A, Becerra D, Corbi-Verge C, Perez-Riba A, Kim PM. Fast and flexible protein design using deep graph neural networks. Cell Syst 2020;11(4):402–11.
[23] Strokach A, Kim PM. Deep generative modeling for protein design. Curr Opin Struct Biol 2022;72:226–36.
[24] Madani A, McCann B, Naik N, Keskar NS, Anand N, Eguchi RR, et al. Progen: language modeling for protein generation. arXiv preprint. arXiv:2004.03497.
[25] Ferruz N, Schmidt S, Höcker B. A deep unsupervised language model for protein design. bioRxiv 2022:2022–03.
[26] Jiménez J, Doerr S, Martínez-Rosell G, Rose AS, De Fabritiis G. Deepsite: protein-binding site predictor using 3d-convolutional neural networks. Bioinformatics 2017;33(19):3036–42.
[27] Gainza P, Sverrisson F, Monti F, Rodola E, Boscaini D, Bronstein M, et al. Deciphering interaction fingerprints from protein molecular surfaces using geometric deep learning. Nat Methods 2020;17(2):184–92.
[28] Zhao J, Cao Y, Zhang L. Exploring the computational methods for protein-ligand binding site prediction. Comput Struct Biotechnol J 2020;18:417–26.
[29] Gomes J, Ramsundar B, Feinberg EN, Pande VS. Atomic convolutional networks for predicting protein-ligand binding affinity. arXiv preprint. arXiv:1703.10603.
[30] Öztürk H, Özgür A, Ozkirimli E. Deepdta: deep drug–target binding affinity prediction. Bioinformatics 2018;34(17):i821–9.
[31] Verma N, Qu X, Trozzi F, Elsaied M, Karki N, Tao Y, et al. Ssnet: a deep learning approach for protein-ligand interaction prediction. Int J Mol Sci 2021;22(3):1392.
[32] Jamasb AR, Day B, Cangea C, Liò P, Blundell TL. Deep learning for protein–protein interaction site prediction. In: Proteomics data analysis. Springer; 2021. p. 263–88.
[33] Soleymani F, Paquet E, Viktor HL, Michalowski W, Spinello D. Protinteract: a deep learning framework for predicting protein–protein interactions. Comput Struct Biotechnol J 2023;21:1324–48.
[34] Dhakal A, McKay C, Tanner JJ, Cheng J. Artificial intelligence in the prediction of protein–ligand interactions: recent advances and future directions. Brief Bioinform 2022;23(1):bbab476.
[35] Jiménez J, Skalic M, Martinez-Rosell G, De Fabritiis G. K_{DEEP}: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. J Chem Inf Model 2018;58(2):287–96.
[36] Wu J, Paquet E, Viktor H, Michalowski W. Protein-protein interaction design with transformers. Available at SSRN 4145752.
[37] Leaver-Fay A, Tyka M, Lewis SM, Lange OF, Thompson J, Jacak R, et al. Rosetta3: an object-oriented software suite for the simulation and design of macromolecules. Methods in enzymology, vol. 487. Elsevier; 2011. p. 545–74.
[38] Anand N, Achim T. Protein structure and sequence generation with equivariant denoising diffusion probabilistic models. arXiv preprint. arXiv:2205.15019.
[39] Ramesh A, Dhariwal P, Nichol A, Chu C, Chen M. Hierarchical text-conditional image generation with clip latents. arXiv preprint. arXiv:2204.06125.

[40] Ramesh A, Pavlov M, Goh G, Gray S, Voss C, Radford A, et al. Zero-shot text-to-image generation. In: International conference on machine learning. PMLR; 2021. p. 8821–31.
[41] Saharia C, Chan W, Saxena S, Li L, Whang J, Denton EL, et al. Photorealistic text-to-image diffusion models with deep language understanding. In: Advances in neural information processing systems, vol. 35. 2022. p. 36479–94.
[42] Watson JL, Juergens D, Bennett NR, Trippe BL, Yim J, Eisenach HE, et al. De novo design of protein structure and function with rfdiffusion. Nature 2023;620(7976):1089–100.
[43] Ingraham J, Garg V, Barzilay R, Jaakkola T. Generative models for graph-based protein design. In: Advances in neural information processing systems, vol. 32. 2019. p. 15820–31.
[44] Trippe BL, Yim J, Tischer D, Broderick T, Baker D, Barzilay R, et al. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. arXiv preprint. arXiv:2206.04119.
[45] Riesselman AJ, Ingraham JB, Marks DS. Deep generative models of genetic variation capture the effects of mutations. Nat Methods 2018;15(10):816–22.
[46] Greener JG, Moffat L, Jones DT. Design of metalloproteins and novel protein folds using variational autoencoders. Sci Rep 2018;8(1):16189.
[47] Rives A, Meier J, Sercu T, Goyal S, Lin Z, Liu J, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proc Natl Acad Sci 2021;118(15):e2016239118.
[48] Lin Z, Sercu T, LeCun Y, Rives A. Deep generative models create new and diverse protein structures. In: Machine learning for structural biology workshop; 2021.
[49] Eguchi RR, Choe CA, Huang P-S. Ig-vae: generative modeling of protein structure by direct 3d coordinate generation. PLoS Comput Biol 2022;18(6):e1010271.
[50] Wu KE, Yang KK, Berg Rvd, Zou JY, Lu AX, Amini AP. Protein structure generation via folding diffusion. arXiv preprint. arXiv:2209.15611.
[51] Sohl-Dickstein J, Weiss E, Maheswaranathan N, Ganguli S. Deep unsupervised learning using nonequilibrium thermodynamics. In: International conference on machine learning. PMLR; 2015. p. 2256–65.
[52] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models. In: Advances in neural information processing systems, vol. 33. 2020. p. 6840–51.
[53] Song J, Meng C, Ermon S. Denoising diffusion implicit models. arXiv preprint. arXiv:2010.02502.
[54] Song Y, Sohl-Dickstein J, Kingma DP, Kumar A, Ermon S, Poole B. Score-based generative modeling through stochastic differential equations. arXiv preprint. arXiv:2011.13456.
[55] Rombach R, Blattmann A, Lorenz D, Esser P, Ommer B. High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2022. p. 10684–95.
[56] Wang Z, Zheng H, He P, Chen W, Zhou M. Diffusion-gan: training gans with diffusion. arXiv preprint. arXiv:2206.02262.
[57] Zheng H, He P, Chen W, Zhou M. Truncated diffusion probabilistic models and diffusion-based adversarial auto-encoders. arXiv preprint. arXiv:2202.09671.
[58] Xie P, Zhang Q, Li Z, Tang H, Du Y, Hu X. Vector quantized diffusion model with codeunet for text-to-sign pose sequences generation. arXiv preprint. arXiv:2208.09141.
[59] Kim D, Kim Y, Kwon SJ, Kang W, Moon I-C. Refining generative process with discriminator guidance in score-based diffusion models. arXiv preprint. arXiv:2211.17091.
[60] Zheng G, Li S, Wang H, Yao T, Chen Y, Ding S, et al. Entropy-driven sampling and training scheme for conditional diffusion generation. In: European conference on computer vision. Springer; 2022. p. 754–69.
[61] Saharia C, Chan W, Chang H, Lee C, Ho J, Salimans T, et al. Palette: image-to-image diffusion models. In: ACM SIGGRAPH 2022 conference proceedings; 2022. p. 1–10.
[62] Wang Y, Yu J, Zhang J. Zero-shot image restoration using denoising diffusion null-space model. arXiv preprint. arXiv:2212.00490.
[63] Lam MW, Wang J, Su D, Yu D. Bddm: bilateral denoising diffusion models for fast and high-quality speech synthesis. arXiv preprint. arXiv:2203.13508.
[64] Li X, Thickstun J, Gulrajani I, Liang PS, Hashimoto TB. Diffusion-lm improves controllable text generation. In: Advances in neural information processing systems, vol. 35. 2022. p. 4328–43.
[65] Austin J, Johnson DD, Ho J, Tarlow D, Van Den Berg R. Structured denoising diffusion models in discrete state-spaces. In: Advances in neural information processing systems, vol. 34. 2021. p. 17981–93.
[66] Hoogeboom E, Nielsen D, Jaini P, Forré P, Welling M. Argmax flows and multinomial diffusion: learning categorical distributions. In: Advances in neural information processing systems, vol. 34. 2021. p. 12454–65.
[67] Savinov N, Chung J, Binkowski M, Elsen E, Oord Avd. Step-unrolled denoising autoencoders for text generation. arXiv preprint. arXiv:2112.06749.
[68] Yu P, Xie S, Ma X, Jia B, Pang B, Gao R, et al. Latent diffusion energy-based model for interpretable text modeling. arXiv preprint. arXiv:2206.05895.
[69] Chen N, Zhang Y, Zen H, Weiss RJ, Norouzi M, Chan W. Wavegrad: estimating gradients for waveform generation. arXiv preprint. arXiv:2009.00713.
[70] Kong Z, Ping W, Huang J, Zhao K, Catanzaro B. Diffwave: a versatile diffusion model for audio synthesis. arXiv preprint. arXiv:2009.09761.
[71] Rasul K, Sheikh A-S, Schuster I, Bergmann U, Vollgraf R. Multivariate probabilistic time series forecasting via conditioned normalizing flows. arXiv preprint. arXiv:2002.06103.
[72] Tashiro Y, Song J, Song Y, Ermon S. Csdi: conditional score-based diffusion models for probabilistic time series imputation. In: Advances in neural information processing systems, vol. 34. 2021. p. 24804–16.
[73] Alcaraz JML, Strodthoff N. Diffusion-based time series imputation and forecasting with structured state space models. arXiv preprint. arXiv:2208.09399.
[74] Avrahami O, Lischinski D, Fried O. Blended diffusion for text-driven editing of natural images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2022. p. 18208–18.
[75] Van den Oord A, Kalchbrenner N, Espeholt L, Vinyals O, Graves A, et al. Conditional image generation with pixelcnn decoders. In: Advances in neural information processing systems, vol. 29. 2016. p. 4797–805.
[76] Papamakarios G, Nalisnick E, Rezende DJ, Mohamed S, Lakshminarayanan B. Normalizing flows for probabilistic modeling and inference. J Mach Learn Res 2021;22(1):2617–80.
[77] LeCun Y, Chopra S, Hadsell R, Ranzato M, Huang F. A tutorial on energy-based learning, Predicting Structured Data 1 (0). http://yann.lecun.com/exdb/publis/orig/lecun-06.pdf.
[78] Kingma DP, Welling M. Auto-encoding variational Bayes. arXiv preprint. arXiv:1312.6114.
[79] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, et al. Generative adversarial networks. Commun ACM 2020;63(11):139–44.
[80] Dhariwal P, Nichol A. Diffusion models beat gans on image synthesis. In: Advances in neural information processing systems, vol. 34. 2021. p. 8780–94.
[81] Li H, Yang Y, Chang M, Chen S, Feng H, Xu Z, et al. Srdiff: single image super-resolution with diffusion probabilistic models. Neurocomputing 2022;479:47–59.
[82] Giannone G, Nielsen D, Winther O. Few-shot diffusion models. arXiv preprint. arXiv:2205.15463.
[83] Lyu Z, Kong Z, Xu X, Pan L, Lin D. A conditional point diffusion-refinement paradigm for 3d point cloud completion. arXiv preprint. arXiv:2112.03530.
[84] Hoogeboom E, Satorras VG, Vignac C, Welling M. Equivariant diffusion for molecule generation in 3d. In: International conference on machine learning. PMLR; 2022. p. 8867–87.
[85] Vahdat A, Kreis K, Kautz J. Score-based generative modeling in latent space. Adv Neural Inf Process Syst 2021;34:11287–302.
[86] Lee JS, Kim J, Kim PM. Proteinsgm: score-based generative modeling for de novo protein design. bioRxiv 2022:2022–07.
[87] Ingraham JB, Baranov M, Costello Z, Barber KW, Wang W, Ismail A, et al. Illuminating protein space with a programmable generative model. Nature 2023:1–9.
[88] Gao Z, Tan C, Li SZ. Diffsds: a language diffusion model for protein backbone inpainting under geometric conditions and constraints. arXiv preprint. arXiv:2301.09642.
[89] Lin Y, AlQuraishi M. Generating novel, designable, and diverse protein structures by equivariantly diffusing oriented residue clouds. arXiv preprint. arXiv:2301.12485.
[90] Yim J, Trippe BL, De Bortoli V, Mathieu E, Doucet A, Barzilay R, et al. Se(3) diffusion model with application to protein backbone generation. arXiv preprint. arXiv:2302.02277.
[91] Luo S, Shi C, Xu M, Tang J. Predicting molecular conformation via dynamic graph score matching. In: Advances in neural information processing systems, vol. 34. 2021. p. 19784–95.
[92] Wu L, Gong C, Liu X, Ye M, Liu Q. Diffusion-based molecule generation with informative prior bridges. In: Advances in neural information processing systems, vol. 35. 2022. p. 36533–45.
[93] Huang H, Sun L, Du B, Lv W. Conditional diffusion based on discrete graph structures for molecular graph generation. arXiv preprint. arXiv:2301.00427.
[94] Zhang H, Li S, Zhang J, Wang Z, Wang J, Jiang D, et al. Sdegen: learning to evolve molecular conformations from thermodynamic noise for conformation generation. Chem Sci 2023;14(6):1557–68.
[95] Wu F, Li SZ. Diffmd: a geometric diffusion model for molecular dynamics simulations. Proceedings of the AAAI conference on artificial intelligence, vol. 37. 2023. p. 5321–9.
[96] Igashov I, Stärk H, Vignac C, Satorras VG, Frossard P, Welling M, et al. Equivariant 3d-conditional diffusion models for molecular linker design. arXiv preprint. arXiv:2210.05274.
[97] Lin H, Huang Y, Liu M, Li X, Ji S, Li SZ. Diffbp: generative diffusion of 3d molecules for target protein binding. arXiv preprint. arXiv:2211.11214.
[98] Schneuing A, Du Y, Harris C, Jamasb A, Igashov I, Du W, et al. Structure-based drug design with equivariant diffusion models. arXiv preprint. arXiv:2210.13695.
[99] Corso G, Stärk H, Jing B, Barzilay R, Jaakkola T. Diffdock: diffusion steps, twists, and turns for molecular docking. arXiv preprint. arXiv:2210.01776.
[100] Qiao Z, Nie W, Vahdat A, Miller III TF, Anandkumar A. Dynamic-backbone protein-ligand structure prediction with multiscale generative diffusion models. arXiv preprint. arXiv:2209.15171.
[101] Valdebenito Maturana CN, Sandoval Orozco AL, García Villalba LJ. Exploration of metrics and datasets to assess the fidelity of images generated by generative adversarial networks. Appl Sci 2023;13(19):10637.
[102] Morales-Juarez E, Fuentes-Pineda G. Efficient generative adversarial networks using linear additive-attention transformers. arXiv preprint. arXiv:2401.09596.
[103] Liang KJ, Li C, Wang G, Carin L. Generative adversarial network training is a continual learning problem. arXiv preprint. arXiv:1811.11083.

[104] Lin L, Liu X, Liang W. Improving variational auto-encoder with self-attention and mutual information for image generation. In: Proceedings of the 3rd international conference on video and image processing; 2019. p. 162–7.
[105] Naderi H, Soleimani BH, Matwin S. Generating high-fidelity images with disentangled adversarial vaes and structure-aware loss. In: 2020 international joint conference on neural networks (IJCNN). IEEE; 2020. p. 1–8.
[106] Miao Y, Yu L, Blunsom P. Neural variational inference for text processing. In: International conference on machine learning. PMLR; 2016. p. 1727–36.
[107] Bastek J-H, Sun W, Kochmann DM. Physics-informed diffusion models. arXiv preprint. arXiv:2403.14404.
[108] Zhang Z, Pi R, Jin Z, Gao Y, Ye J, Chen K, et al. Efficient denoising diffusion via probabilistic masking.
[109] Cao H, Tan C, Gao Z, Xu Y, Chen G, Heng P-A, et al. A survey on generative diffusion models. IEEE Trans Knowl Data Eng 2024;36(7):2814–30.
[110] Croitoru F-A, Hondru V, Ionescu RT, Shah M. Diffusion models in vision: A survey. IEEE Trans Pattern Anal Mach Intell 2023;45(9):10850–69.
[111] Kuhlman B, Bradley P. Advances in protein structure prediction and design. Nat Rev Mol Cell Biol 2019;20(11):681–97.
[112] Huang P-S, Boyken SE, Baker D. The coming of age of de novo protein design. Nature 2016;537(7620):320–7.
[113] Krissinel E. On the relationship between sequence and structure similarities in proteomics. Bioinformatics 2007;23(6):717–23.
[114] Maynard Smith J. Natural selection and the concept of a protein space. Nature 1970;225(5232):563–4.
[115] Goverde CA, Wolf B, Khakzad H, Rosset S, Correia BE. De novo protein design by inversion of the alphafold structure prediction network. Protein Sci 2023;32(6):e4653.
[116] Huang P-S, Feldmeier K, Parmeggiani F, Fernandez Velasco DA, Höcker B, Baker D. De novo design of a four-fold symmetric tim-barrel protein with atomic-level accuracy. Nat Chem Biol 2016;12(1):29–34.
[117] Dou J, Vorobieva AA, Sheffler W, Doyle LA, Park H, Bick MJ, et al. De novo design of a fluorescence-activating β-barrel. Nature 2018;561(7724):485–91.
[118] Silva D-A, Yu S, Ulge UY, Spangler JB, Jude KM, Labão-Almeida C, et al. De novo design of potent and selective mimics of il-2 and il-15. Nature 2019;565(7738):186–91.
[119] Langan RA, Boyken SE, Ng AH, Samson JA, Dods G, Westbrook AM, et al. De novo design of bioactive protein switches. Nature 2019;572(7768):205–10.
[120] Wei KY, Moschidi D, Bick MJ, Nerli S, McShan AC, Carter LP, et al. Computational design of closely related proteins that adopt two well-defined but structurally divergent folds. Proc Natl Acad Sci 2020;117(13):7208–15.
[121] Xu M, Yu L, Song Y, Shi C, Ermon S, Tang J. Geodiff: a geometric diffusion model for molecular conformation generation. arXiv preprint. arXiv:2203.02923.
[122] Eguchi RR, Anand N, Choe CA, Huang P-S. Ig-vae: generative modeling of immunoglobulin proteins by direct 3d coordinate generation. bioRxiv 2020;2020:8.
[123] Lodish H, Zipursky SL. Molecular cell biology. Biochem Mol Biol Educ 2001;29:126–33.
[124] Camilloni C, Bonetti D, Morrone A, Giri R, Dobson CM, Brunori M, et al. Towards a structural biology of the hydrophobic effect in protein folding. Sci Rep 2016;6(1):1–9.
[125] Alberts B. Molecular biology of the cell. Garland Science; 2017.
[126] Berkholz DS, Shapovalov MV, Dunbrack RL, Karplus PA. Conformation dependence of backbone geometry in proteins. Structure 2009;17(10):1316–25.
[127] Zheng X, Gan L, Wang E, Wang J. Pocket-based drug design: exploring pocket space. AAPS J 2013;15:228–41.
[128] Nivedha S, Bhavani S. A survey on prediction of protein-protein interactions. Journal of physics: conference series, vol. 1937. IOP Publishing; 2021. p. 012011.
[129] Bepler T, Berger B. Learning protein sequence embeddings using information from structure. arXiv preprint. arXiv:1902.08661.
[130] Wang S, Sun S, Li Z, Zhang R, Xu J. Accurate de novo prediction of protein contact map by ultra-deep learning model. PLoS Comput Biol 2017;13(1):e1005324.
[131] Kulmanov M, Khan MA, Hoehndorf R. Deepgo: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. Bioinformatics 2018;34(4):660–8.
[132] Gligorijević V, Renfrew PD, Kosciolek T, Leman JK, Berenberg D, Vatanen T, et al. Structure-based protein function prediction using graph convolutional networks. Nat Commun 2021;12(1):3168.
[133] Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. arXiv preprint. arXiv:1609.02907.
[134] Bruna J, Zaremba W, Szlam A, LeCun Y. Spectral networks and locally connected networks on graphs. arXiv preprint. arXiv:1312.6203.
[135] Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in neural information processing systems, vol. 29. 2016. p. 3844–52.
[136] Ding W, Nakai K, Gong H. Protein design via deep learning. Brief Bioinform 2022;23(3):bbac102.
[137] Réau M, Renaud N, Xue LC, Bonvin AM. Deeprank-gnn: a graph neural network framework to learn patterns in protein–protein interfaces. Bioinformatics 2023;39(1):btac759.
[138] Lee M. Recent advances in deep learning for protein-protein interaction analysis: a comprehensive review. Molecules 2023;28(13):5169.
[139] Weiler M, Cesa G. General e(2)-equivariant steerable cnns. In: Advances in neural information processing systems, vol. 32. 2019. p. 14334–45.
[140] Rezende DJ, Racanière S, Higgins I, Toth P. Equivariant Hamiltonian flows. arXiv preprint. arXiv:1909.13739.
[141] Fasoulis R, Paliouras G, Kavraki LE. Graph representation learning for structural proteomics. Emerg Top Life Sci 2021;5(6):789–802.
[142] Henaff M, Bruna J, LeCun Y. Deep convolutional networks on graph-structured data. arXiv preprint. arXiv:1506.05163.
[143] Bronstein MM, Bruna J, LeCun Y, Szlam A, Vandergheynst P. Geometric deep learning: going beyond Euclidean data. IEEE Signal Process Mag 2017;34(4):18–42.
[144] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. Pytorch: an imperative style, high-performance deep learning library. In: Advances in neural information processing systems, vol. 32. 2019. p. 8024–35.
[145] Fout A, Byrd J, Shariat B, Ben-Hur A. Protein interface prediction using graph convolutional networks. In: Advances in neural information processing systems, vol. 30. 2017. p. 6530–9.
[146] Veličković P, Cucurull G, Casanova A, Romero A, Lio P, Bengio Y. Graph attention networks. arXiv preprint. arXiv:1710.10903.
[147] Huang X, Song Q, Li Y, Hu X. Graph recurrent networks with attributed random walks. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining; 2019. p. 732–40.
[148] Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, et al. Graph neural networks: a review of methods and applications. AI Open 2020;1:57–81.
[149] Ying Z, You J, Morris C, Ren X, Hamilton W, Leskovec J. Hierarchical graph representation learning with differentiable pooling. In: Advances in neural information processing systems, vol. 31. 2018. p. 4800–10.
[150] Gao H, Ji S. Graph u-nets. In: International conference on machine learning. PMLR; 2019. p. 2083–92.
[151] Li S, Zhou J, Xu T, Huang L, Wang F, Xiong H, et al. Structure-aware interactive graph neural networks for the prediction of protein-ligand binding affinity. In: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining; 2021. p. 975–85.
[152] Del Vecchio A, Deac A, Liò P, Veličković P. Neural message passing for joint paratope-epitope prediction. arXiv preprint. arXiv:2106.00757.
[153] Fu C, Yan K, Wang L, Au WY, McThrow MC, Komikado T, et al. A latent diffusion model for protein structure generation. In: Learning on graphs conference. PMLR; 2024:29.
[154] Gruver N, Stanton S, Frey N, Rudner TG, Hotzel I, Lafrance-Vanasse J, et al. Protein design with guided discrete diffusion. Adv Neural Inf Process Syst 2023;36.
[155] Chen M, Mei S, Fan J, Wang M. An overview of diffusion models: applications, guided generation, statistical rates and optimization. arXiv preprint. arXiv:2404.07771.
[156] Brock A, Donahue J, Simonyan K. Large scale gan training for high fidelity natural image synthesis. arXiv preprint. arXiv:1809.11096.
[157] Du Y, Mordatch I. Implicit generation and modeling with energy based models. In: Advances in neural information processing systems, vol. 32. 2019. p. 3608–18.
[158] Karras T, Aittala M, Hellsten J, Laine S, Lehtinen J, Aila T. Training generative adversarial networks with limited data. In: Advances in neural information processing systems, vol. 33. 2020. p. 12104–14.
[159] Rezende DJ, Mohamed S, Wierstra D. Stochastic backpropagation and approximate inference in deep generative models. In: International conference on machine learning. PMLR; 2014. p. 1278–86.
[160] Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. In: Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings, part III 18. Springer; 2015. p. 234–41.
[161] Xu M, Powers AS, Dror RO, Ermon S, Leskovec J. Geometric latent diffusion models for 3d molecule generation. In: International conference on machine learning. PMLR; 2023. p. 38592–610.
[162] Defresne M, Barbe S, Schiex T. Protein design with deep learning. Int J Mol Sci 2021;22(21):11741.
[163] Liu M, Luo Y, Uchino K, Maruhashi K, Ji S. Generating 3d molecules for target protein binding. arXiv preprint. arXiv:2204.09410.
[164] Farina F, Slade E. Symmetry-driven graph neural networks. arXiv preprint. arXiv:2105.14058.
[165] Brandstetter J, Hesselink R, van der Pol E, Bekkers EJ, Welling M. Geometric and physical quantities improve e(3) equivariant message passing. arXiv preprint. arXiv:2110.02905.
[166] Roche R, Moussad B, Shuvo MH, Bhattacharya D. E (3) equivariant graph neural networks for robust and accurate protein-protein interaction site prediction. PLoS Comput Biol 2023;19(8):e1011435.
[167] Nelson DL, Lehninger AL, Cox MM. Lehninger principles of biochemistry. Macmillan; 2008.
[168] Hall BC. Lie groups, Lie algebras, and representations. Springer; 2013.
[169] Unke OT, Maennel H. E3x: E(3)-equivariant deep learning made easy. arXiv preprint. arXiv:2401.07595.
[170] Li R-W, Zhang L-X, Li C, Lai Y-K, Gao L. E3sym: leveraging e (3) invariance for unsupervised 3d planar reflective symmetry detection. In: Proceedings of the IEEE/CVF international conference on computer vision; 2023. p. 14543–53.
[171] Brooks WH, Guida WC, Daniel KG. The significance of chirality in drug design and development. Curr Top Med Chem 2011;11(7):760–70.

[172] Inaki M, Liu J, Matsuno K. Cell chirality: its origin and roles in left–right asymmetric development. Philos Trans R Soc Lond B, Biol Sci 2016;371(1710):20150403.
[173] Guo A-M, Sun Q-F. Spin-dependent electron transport in protein-like single-helical molecules. Proc Natl Acad Sci 2014;111(32):11658–62.
[174] Satorras VG, Hoogeboom E, Welling M. E(n) equivariant graph neural networks. In: International conference on machine learning. PMLR; 2021. p. 9323–32.
[175] Rahman T, Alam FF, Shehu A. Equivariant encoding based gvae (eqen-gvae) for protein tertiary structure generation. In: 2022 IEEE international conference on bioinformatics and biomedicine (BIBM). IEEE; 2022. p. 3470–7.
[176] Dhakal A, Gyawali R, Cheng J. Predicting protein-ligand binding structure using e(n) equivariant graph neural networks. bioRxiv 2023:2023–08.
[177] Jing B, Eismann S, Soni PN, Dror RO. Equivariant graph neural networks for 3d macromolecular structure. arXiv preprint. arXiv:2106.03843.
[178] Leach A, Schmon SM, Degiacomi MT, Willcocks CG. Denoising diffusion probabilistic models on so(3) for rotational alignment. In: ICLR 2022 workshop on geometrical and topological representation learning; 2022. p. 53.
[179] Mohamed S, Lakshminarayanan B. Learning in implicit generative models. arXiv preprint. arXiv:1610.03483.
[180] Zhang G, Ji J, Zhang Y, Yu M, Jaakkola T, Chang S. Towards coherent image inpainting using denoising diffusion implicit models. In: International conference on machine learning. PMLR; 2023. p. 41164–93.
[181] Yang L, Zhang Z, Song Y, Hong S, Xu R, Zhao Y, et al. Diffusion models: a comprehensive survey of methods and applications. arXiv preprint. arXiv:2209.00796.
[182] Han J, Rong Y, Xu T, Huang W. Geometrically equivariant graph neural networks: a survey. arXiv preprint. arXiv:2202.07230.
[183] Li Y, Wang Y, Huang L, Yang H, Wei X, Zhang J, et al. Long-short-range message-passing: a physics-informed framework to capture non-local interaction for scalable molecular dynamics simulation. arXiv preprint. arXiv:2304.13542.
[184] Derevyanko G, Grudinin S, Bengio Y, Lamoureux G. Deep convolutional networks for quality assessment of protein folds. Bioinformatics 2018;34(23):4046–53.
[185] Jamasb AR, Viñas R, Ma EJ, Harris C, Huang K, Hall D, et al. Graphein-a python library for geometric deep learning and network analysis on protein structures and interaction networks. bioRxiv 2020:2020–07.
[186] Aluri M, Tatavarthi UD. Geometric deep learning for enhancing irregular scene text detection. Rev Intell Artif 38(1).
[187] Chen C, Chen X, Morehead A, Wu T, Cheng J. 3d-equivariant graph neural networks for protein model quality assessment. Bioinformatics 2023;39(1):btad030.
[188] Garcia Satorras V, Hoogeboom E, Fuchs F, Posner I, Welling M. E(n) equivariant normalizing flows. In: Advances in neural information processing systems, vol. 34. 2021. p. 4181–92.
[189] Maximova T, Moffatt R, Ma B, Nussinov R, Shehu A. Principles and overview of sampling methods for modeling macromolecular structure and dynamics. PLoS Comput Biol 2016;12(4):e1004619.
[190] Clausen R, Ma B, Nussinov R, Shehu A. Mapping the conformation space of wild-type and mutant h-ras with a memetic, cellular, and multiscale evolutionary algorithm. PLoS Comput Biol 2015;11(9):e1004470.
[191] Sapin E, Carr DB, De Jong KA, Shehu A. Computing energy landscape maps and structural excursions of proteins. BMC Genomics 2016;17(4):433–56.
[192] Maximova T, Plaku E, Shehu A. Structure-guided protein transition modeling with a probabilistic roadmap algorithm. IEEE/ACM Trans Comput Biol Bioinform 2016;15(6):1783–96.
[193] Anand N, Eguchi R, Huang P-S. Fully differentiable full-atom protein backbone generation. In: ICLR 2019 workshop; 2019. p. 35.
[194] Ovchinnikov S, Huang P-S. Structure-based protein design with deep learning. Curr Opin Chem Biol 2021;65:136–44.
[195] Huang H, Amor BB, Lin X, Zhu F, Fang Y. G-vae, a geometric convolutional vae for protein structure generation. arXiv preprint. arXiv:2106.11920.
[196] Guo X, Du Y, Tadepalli S, Zhao L, Shehu A. Generating tertiary protein structures via an interpretative variational autoencoder. arXiv preprint. arXiv:2004.07119.
[197] Simonovsky M, Komodakis N. Graphvae: towards generation of small graphs using variational autoencoders. In: Artificial neural networks and machine learning–ICANN 2018: 27th international conference on artificial neural networks, Rhodes, Greece, October 4–7, 2018, proceedings, part I 27. Springer; 2018. p. 412–22.
[198] Hoffmann M, Noé F. Generating valid Euclidean distance matrices. arXiv preprint. arXiv:1910.03131.
[199] Hamilton WL, Ying R, Leskovec J. Representation learning on graphs: methods and applications. arXiv preprint. arXiv:1709.05584.
[200] De Cao N, Kipf T. Molgan: an implicit generative model for small molecular graphs. arXiv preprint. arXiv:1805.11973.
[201] Liu W, Chen P-Y, Yu F, Suzumura T, Hu G. Learning graph topological features via gan. IEEE Access 2019;7:21834–43.
[202] Köhler J, Klein L, Noé F. Equivariant flows: exact likelihood generative learning for symmetric densities. In: International conference on machine learning. PMLR; 2020. p. 5361–70.
[203] Luo Y, Yan K, Ji S. Graphdf: a discrete flow model for molecular graph generation. In: International conference on machine learning. PMLR; 2021. p. 7192–203.
[204] Jing B, Corso G, Chang J, Barzilay R, Jaakkola T. Torsional diffusion for molecular conformer generation. In: Advances in neural information processing systems, vol. 35. 2022. p. 24240–53.
[205] Gebauer N, Gastegger M, Schütt K. Symmetry-adapted generation of 3d point sets for the targeted discovery of molecules. In: Advances in neural information processing systems, vol. 32. 2019. p. 7566–78.
[206] Seo A, Kim B, Kwak S, Cho M. Reflection and rotation symmetry detection via equivariant learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2022. p. 9539–48.
[207] Hoseini P, Zhao L, Shehu A. Generative deep learning for macromolecular structure and dynamics. Curr Opin Struct Biol 2021;67:170–7.
[208] Gilmer J, Schoenholz SS, Riley PF, Vinyals O, Dahl GE. Neural message passing for quantum chemistry. In: International conference on machine learning. PMLR; 2017. p. 1263–72.
[209] Battaglia PW, Hamrick JB, Bapst V, Sanchez-Gonzalez A, Zambaldi V, Malinowski M, et al. Relational inductive biases, deep learning, and graph networks. arXiv preprint. arXiv:1806.01261.
[210] Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, et al. The protein data bank. Nucleic Acids Res 2000;28(1):235–42.
[211] De Bortoli V, Mathieu E, Hutchinson M, Thornton J, Teh YW, Doucet A. Riemannian score-based generative modelling. In: Advances in neural information processing systems, vol. 35. 2022. p. 2406–22.
[212] Dauparas J, Anishchenko I, Bennett N, Bai H, Ragotte RJ, Milles LF, et al. Robust deep learning–based protein sequence design using proteinmpnn. Science 2022;378(6615):49–56.
[213] Wang J, Lisanza S, Juergens D, Tischer D, Watson JL, Castro KM, et al. Scaffolding protein functional sites using deep learning. Science 2022;377(6604):387–94.
[214] Barnes J, Hut P. A hierarchical O(N log N) force-calculation algorithm. Nature 1986;324(6096):446–9.
[215] Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. bioRxiv 2022;2022:500902.
[216] Hu S, Lundgren M, Niemi AJ. Discrete Frenet frame, inflection point solitons, and curve visualization with applications to folded proteins. Phys Rev E 2011;83(6):061908.
[217] Peng X, Luo S, Guan J, Xie Q, Peng J, Ma J. Pocket2mol: efficient molecular sampling based on 3d protein pockets. In: International conference on machine learning. PMLR; 2022. p. 17644–55.
[218] Shi C, Luo S, Xu M, Tang J. Learning gradient fields for molecular conformation generation. In: International conference on machine learning. PMLR; 2021. p. 9558–68.
[219] Yim J, Stärk H, Corso G, Jing B, Barzilay R, Jaakkola TS. Diffusion models in protein structure and docking. Wiley Interdiscip Rev Comput Mol Sci 2024;14(2):e1711.
[220] Gao W, Mahajan SP, Sulam J, Gray JJ. Deep learning in protein structural modeling and design. Patterns 1(9).
[221] Zhang M, Qamar M, Kang T, Jung Y, Zhang C, Bae S-H, et al. A survey on graph diffusion models: generative AI in science for molecule, protein and material. arXiv preprint. arXiv:2304.01565.
[222] Ramakrishnan R, Dral PO, Rupp M, Von Lilienfeld OA. Quantum chemistry structures and properties of 134 kilo molecules. Sci Data 2014;1(1):1–7.
[223] Ruddigkeit L, Van Deursen R, Blum LC, Reymond J-L. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. J Chem Inf Model 2012;52(11):2864–75.
[224] Sillitoe I, Bordin N, Dawson N, Waman VP, Ashford P, Scholes HM, et al. Cath: increased structural coverage of functional space. Nucleic Acids Res 2021;49(D1):D266–73.
[225] Todd AE, Orengo CA, Thornton JM. Evolution of function in protein superfamilies, from a structural perspective. J Mol Biol 2001;307(4):1113–43.
[226] Pearl FM, Bennett C, Bray JE, Harrison AP, Martin N, Shepherd A, et al. The cath database: an extended protein family resource for structural and functional genomics. Nucleic Acids Res 2003;31(1):452–5.
[227] Sillitoe I, Dawson N, Lewis TE, Das S, Lees JG, Ashford P, et al. Cath: expanding the horizons of structure-based functional annotations for genome sequences. Nucleic Acids Res 2019;47(D1):D280–4.
[228] Wu Z, Ramsundar B, Feinberg EN, Gomes J, Geniesse C, Pappu AS, et al. Moleculenet: a benchmark for molecular machine learning. Chem Sci 2018;9(2):513–30.
[229] Pracht P, Bohle F, Grimme S. Automated exploration of the low-energy chemical space with fast quantum chemical methods. Phys Chem Chem Phys 2020;22(14):7169–92.
[230] Bannwarth C, Ehlert S, Grimme S. Gfn2-xtb—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions. J Chem Theory Comput 2019;15(3):1652–71.
[231] Subramanian G, Ramsundar B, Pande V, Denny RA. Computational modeling of β-secretase 1 (bace-1) inhibitors using ligand based approaches. J Chem Inf Model 2016;56(10):1936–49.
[232] Burley SK, Berman HM, Kleywegt GJ, Markley JL, Nakamura H, Velankar S. Protein data bank (pdb): the single global macromolecular structure archive. In: Protein crystallography: methods and protocols; 2017. p. 627–41.
[233] Rose Y, Duarte JM, Lowe R, Segura J, Bi C, Bhikadiya C, et al. Rcsb protein data bank: architectural advances towards integrated searching and efficient access to macromolecular structure data from the pdb archive. J Mol Biol 2021;433(11):166704.

[234] Burley SK, Bhikadiya C, Bi C, Bittrich S, Chao H, Chen L, et al. Rcsb protein data bank (rcsb.org): delivery of experimentally-determined pdb structures alongside one million computed structure models of proteins from artificial intelligence/machine learning. Nucleic Acids Res 2023;51(D1):D488–508.
[235] Varadi M, Anyango S, Deshpande M, Nair S, Natassia C, Yordanova G, et al. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. Nucleic Acids Res 2022;50(D1):D439–44.
[236] The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2023. Nucleic Acids Res 2023;51(D1):D523–31.
[237] Morales J, Pujar S, Loveland JE, Astashyn A, Bennett R, Berry A, et al. A joint ncbi and embl-ebi transcript set for clinical genomics and research. Nature 2022;604(7905):310–5.
[238] Humphreys IR, Pei J, Baek M, Krishnakumar A, Anishchenko I, Ovchinnikov S, et al. Computed structures of core eukaryotic protein complexes. Science 2021;374(6573):eabm4805.
[239] Anderson B, Hy TS, Kondor R. Cormorant: covariant molecular neural networks. In: Advances in neural information processing systems, vol. 32. 2019. p. 14537–46.
[240] Dawson NL, Lewis TE, Das S, Lees JG, Lee D, Ashford P, et al. Cath: an expanded resource to predict protein function through structure and sequence. Nucleic Acids Res 2017;45(D1):D289–95.
[241] Lewis TE, Sillitoe I, Dawson N, Lam SD, Clarke T, Lee D, et al. Gene3d: extensive prediction of globular domains in proteins. Nucleic Acids Res 2018;46(D1):D435–9.
[242] Sillitoe I, Lewis TE, Cuff A, Das S, Ashford P, Dawson NL, et al. Cath: comprehensive structural and functional annotations for genome sequences. Nucleic Acids Res 2015;43(D1):D376–81.
[243] Axelrod S, Gomez-Bombarelli R. Geom, energy-annotated molecular conformations for property prediction and molecular generation. Sci Data 2022;9(1):185.
[244] Bhardwaj A, Kaur J, Wuest M, Wuest F. In situ click chemistry generation of cyclooxygenase-2 inhibitors. Nat Commun 2017;8(1):1.
[245] Fox NK, Brenner SE, Chandonia J-M. Scope: structural classification of proteins—extended, integrating scop and astral data and classification of new structures. Nucleic Acids Res 2014;42(D1):D304–9.
[246] Chandonia J-M, Guan L, Lin S, Yu C, Fox NK, Brenner SE. Scope: improvements to the structural classification of proteins–extended database to facilitate variant interpretation and machine learning. Nucleic Acids Res 2022;50(D1):D553–9.
[247] Lim R. Methods for accelerating machine learning in high performance computing. University of Oregon—Area-2019-01.
[248] Watson JL, Juergens D, Bennett NR, Trippe BL, Yim J, Eisenach HE, et al. Broadly applicable and accurate protein design by integrating structure prediction networks and diffusion generative models. bioRxiv 2022:2022–12.
[249] Chen T, Hong L, Yudistyra V, Vincoff S, Chatterjee P. Generative design of therapeutics that bind and modulate protein states. Curr Opin Biomed Eng 2023:100496.
[250] Feng S, Sáez M, Wiuf C, Feliu E, Soyer OS. Core signalling motif displaying multistability through multi-state enzymes. J R Soc Interface 2016;13(123):20160524.